

Using GAMS STUDIO



Bruce A. McCarl

Specialist in Applied Optimization
Distinguished Professor of Agricultural Economics,
Texas A&M University
Principal, McCarl and Associates

mccarl@tamu.edu

brucemccarl@gmail.com

<http://agecon2.tamu.edu/people/faculty/mccarl-bruce/>

Using GAMS STUDIO What Is It?

A multi platform (PCs with Windows, Mac, LINUX)
graphical interface to run GAMS



GAMS uses two phases.

1- Text edit

First, one uses a **text editor** and creates a file which contains GAMS instructions.

2- File submission

Second, one **submits that file** to GAMS which executes those instructions causing calculations to be done, solvers to be used and a solution file of the execution results to be created.

Approaches to use GAMS.

1. More batch oriented Traditional method
use a text editor set up the model then use **DOS (or LINUX)** command line instructions to find errors in and run the model.
2. GAMS STUDIO or GAMS IDE alternatives.
Graphical interfaces to create, debug, text edit and run GAMS files.

Using GAMS STUDIO

Summary of steps to using

1. Install GAMS and STUDIO on your computer



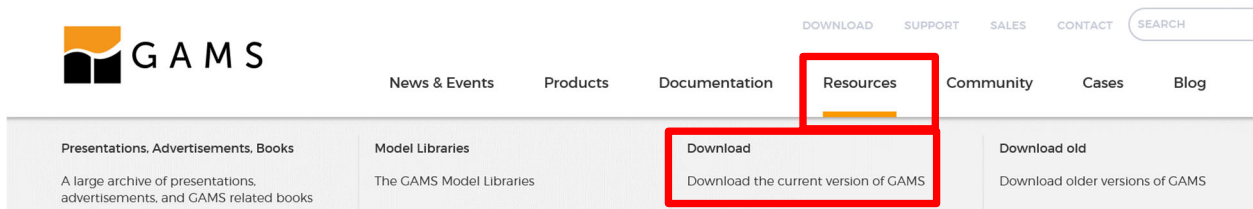
2. Open STUDIO through the icon
3. Open or create a file by going to the file selection in the upper left corner.
4. Prepare the file so you think it is ready for execution
5. Run the file with GAMS by clicking the run button or pressing F9
6. Open and navigate around the output

Using GAMS STUDIO Installation

Install GAMS and STUDIO

GAMS STUDIO is **automatically installed** when GAMS is installed and you get the **choice of whether to use STUDIO or the IDE as primary editor**. To install whole GAMS system including STUDIO do the following steps

- a. Download the GAMS installation file from www.GAMS.com under Resources and Download



- b. Start the installation

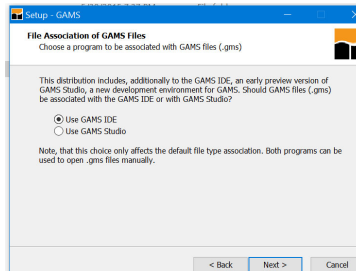
Installer should place an icon on the desktop yielding



Using GAMS STUDIO

Using STUDIO after Installation

During the install you will find

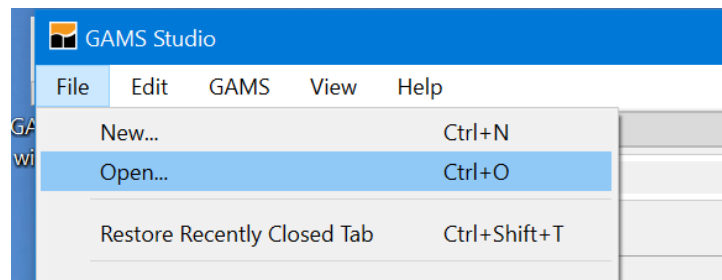


Which allows you to choose which graphical interface to use as the default. For now choose STUDIO.

Open STUDIO through the icon



Open the File menu choice. Select a file.

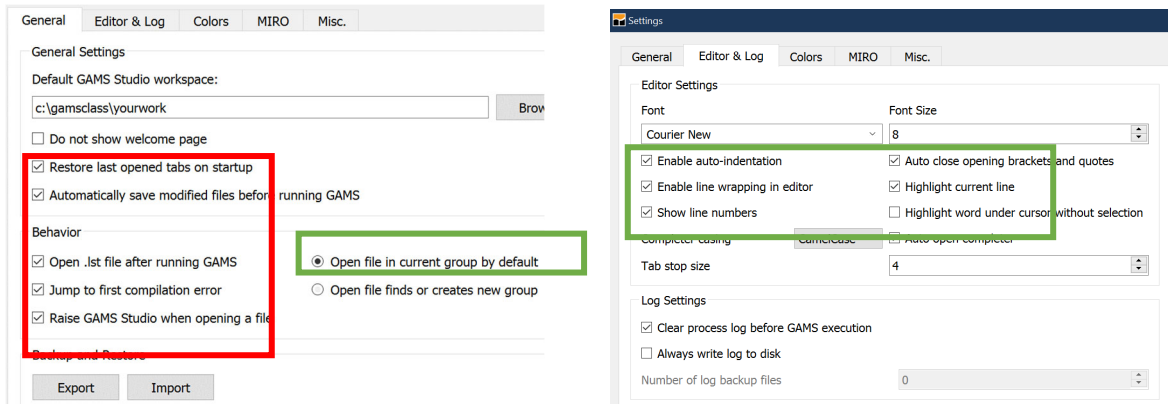


Note, the file location determines where all generated files are placed (to place files elsewhere use the save as dialogue) and where GAMS looks for files when executing. I recommend that you make sure your file is in a place where you want it also making sure that place contains all associated files i.e. this is the file storage directory for this endeavor.

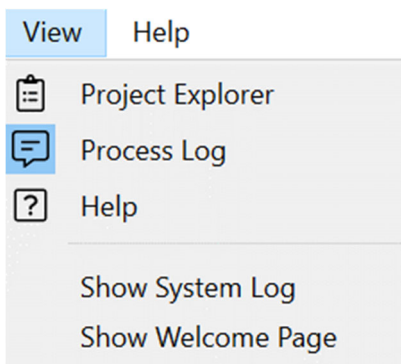
Using GAMS STUDIO

Using STUDIO after Installation

Fix defaults and suppress the highlighting of word under the cursor (I find it is very distracting). When first opening STUDIO go to settings and I would make sure check marks are as follows especially the checked ones in the **red boxes** and I would choose the **green** ones



You should also select what windows that you want to see. Under view you can activate the explorer and process log

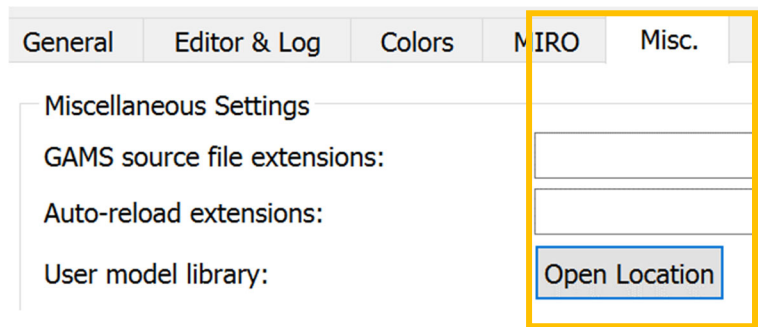


Here selecting the process log window and possibly the Explorer are of interest. The **Explorer** will open automatically when a file is opened and the **Process Log window** will open when a solve is done. I usually hide the Explorer and sometimes want to get rid of the Process window.

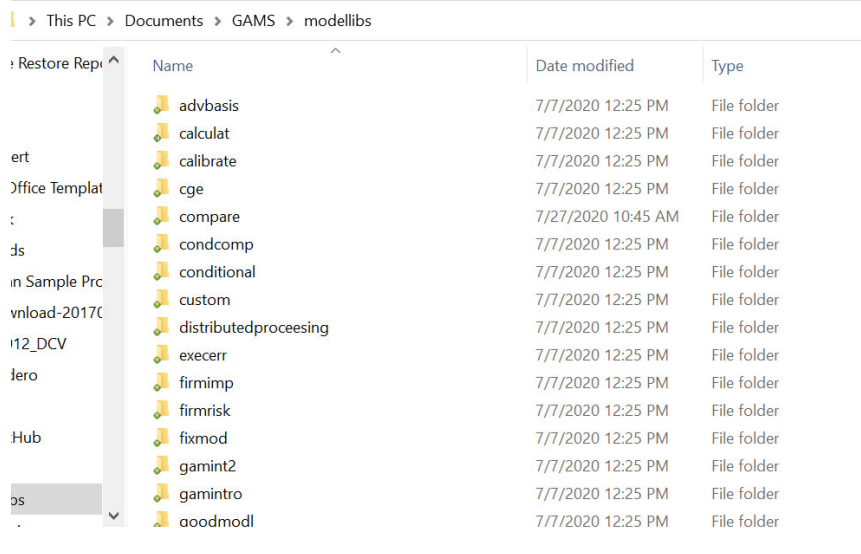
Using GAMS STUDIO

Using STUDIO after Installation

You should also add the mccarl library that is in the file `c:\gamsclass\examples\modlib.glb`. to do this we need to copy that directory to the location identified in the **Misc tab**



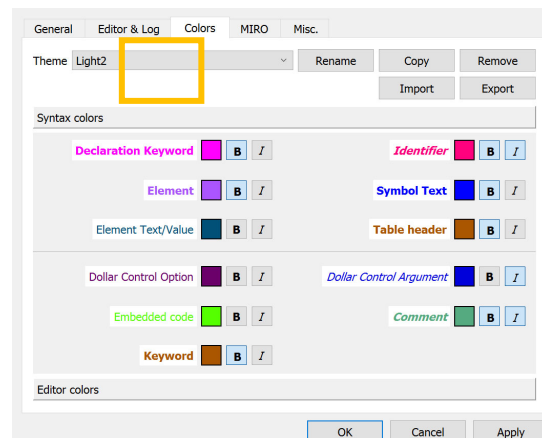
When that is clicked you should see the following if the install was done properly



Note the library is stored on a windows machine in `C:\Users\mccarl\Documents\GAMS\modellibs`

Using GAMS STUDIO Using STUDIO after Installation

You can also manipulate the syntax coloring being used in STUDIO. Go to settings and select COLORS and the window below will show up



In turn you can manipulate the text coloring, bolding and italicizing in STUDIO for **GAMS declaration keywords** (*set, parameter, Variables etc.*), **other keywords** (*set, solve etc*), **elements in sets**, and other things.

I especially do this for comments as it shows me active and inactive code.

Using GAMS STUDIO

Getting Started

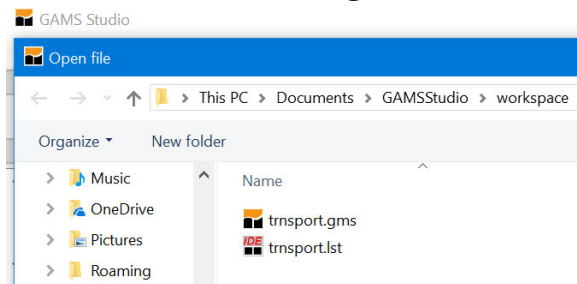
Create or open an existing GMS file

Several cases are possible

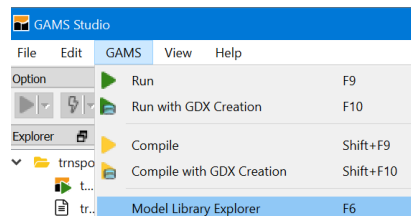
- Create a new file (covered later)
- Open an existing file – at first this will be in

» This PC » Documents » GAMSSStudio » workspace

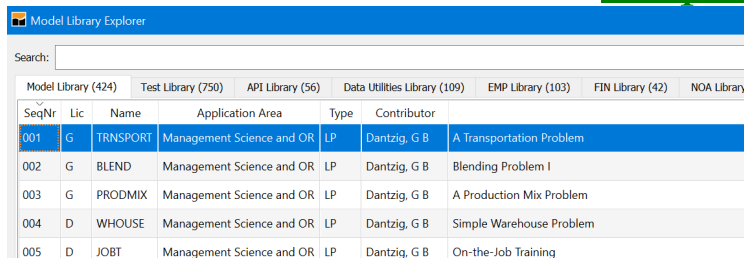
then you see something like



- Open the GAMS model library explorer under the GAMS tab or press F6 (this is the simplest for first time users and the one we will use)



Select a model like [trnsport.gms](#)

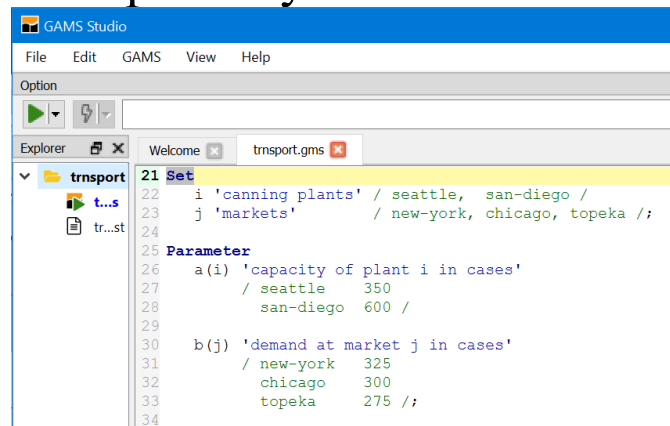


SeqNr	Lic	Name	Application Area	Type	Contributor	
001	G	TRANSPORT	Management Science and OR	LP	Dantzig, G B	A Transportation Problem
002	G	BLEND	Management Science and OR	LP	Dantzig, G B	Blending Problem I
003	G	PRODMIX	Management Science and OR	LP	Dantzig, G B	A Production Mix Problem
004	D	WHOUSE	Management Science and OR	LP	Dantzig, G B	Simple Warehouse Problem
005	D	JOBT	Management Science and OR	LP	Dantzig, G B	On-the-Job Training

It will be automatically saved in the last directory you used or in the STUDIO default Workspace directory.

Using GAMS STUDIO Prepare file for execution

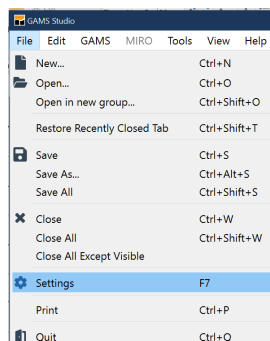
When using model library [trnsport.gms](#) should now appear as part of your STUDIO screen



```
21 Set
22 i 'canning plants' / seattle, san-diego /
23 j 'markets' / new-york, chicago, topeka /;
24
25 Parameter
26 a(i) 'capacity of plant i in cases'
27 / seattle 350
28 san-diego 600 /
29
30 b(j) 'demand at market j in cases'
31 / new-york 325
32 chicago 300
33 topeka 275 /;
34
```

The STUDIO contains a full-featured editor. Go through the file and change what you want.

You can also customize the appearance somewhat through the settings choice under the File menu



There you can change font size (under editor), tabs, default file location (called workspace) and some other features on how GAMS runs. For now leave it alone

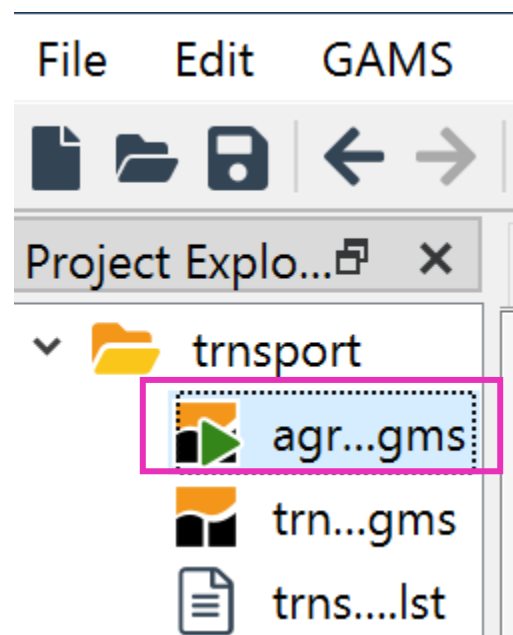
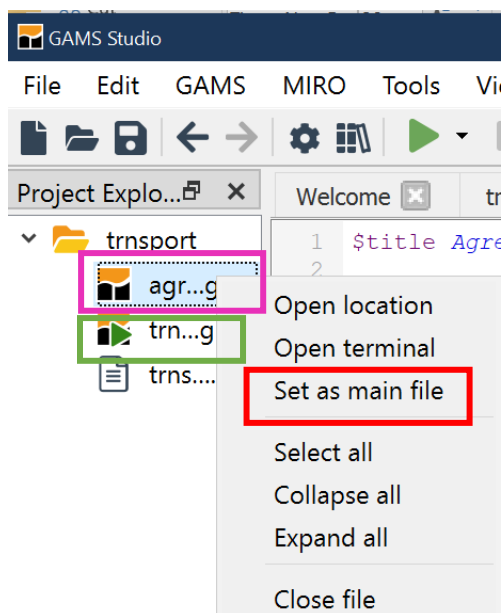
Using GAMS STUDIO

Getting Started

Make sure right file is executed

Unlike many other programs, STUDIO does not necessarily execute the file you have been working on rather executing what it calls the "main" file.

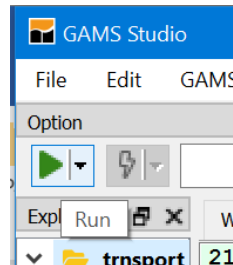
In particular, if you activate the explorer view then the main file is the one marked with a **green triangle** as below. If the triangle is not marking the file to be run you need to right **click on that file** and then choose **Set as main file**. In turn the triangle will **move**.



Using GAMS STUDIO

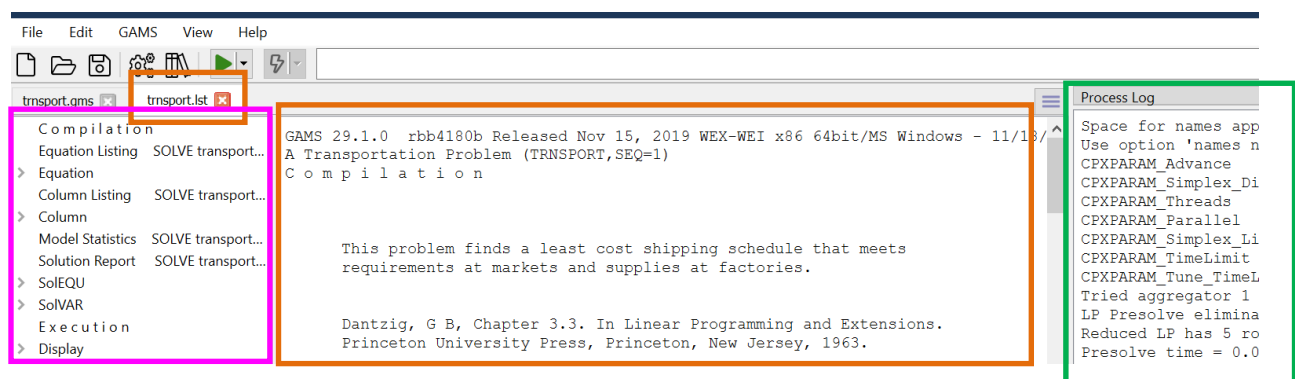
Getting Started

Run GAMS by clicking the run  button



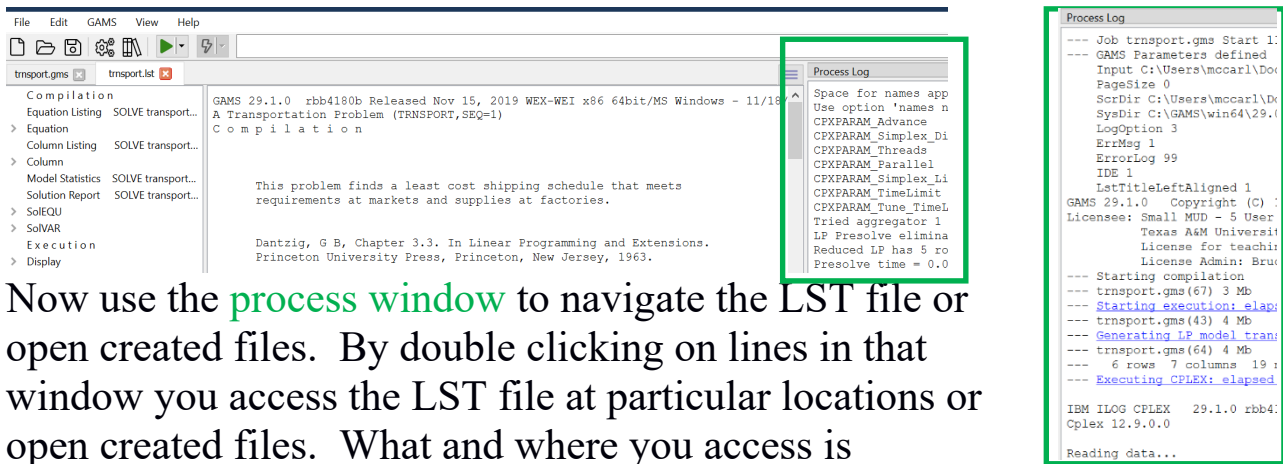
The so called **Process log** window will then appear which logs the steps executed in running the model. The **LST file** and the **navigation window** also appear. Note to save space I would close the Explorer window and drag the **Process Log** window to the bottom or the right hand side.

The Process Log window on the right is illustrated here and it is on the right side on the next page.



Using GAMS STUDIO Getting Started

Here it is after reconfiguring by dragging the **process Log window** up and to the right then resizing and repositioning



Now use the **process window** to navigate the LST file or open created files. By double clicking on lines in that window you access the LST file at particular locations or open created files. What and where you access is determined by the color of the line you click on

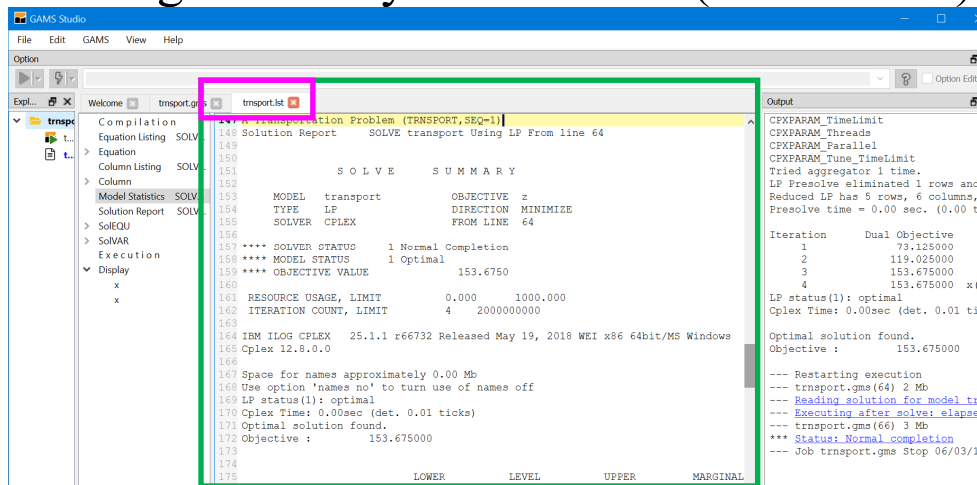
Color of Line	Function and Destination When Double Clicked
Blue line	Opens LST file and jumps to corresponding place. Blue lines also open put and other files created in run.
Non-bolded black line	Control click on this takes you to LST file. Coble click on error message takes you to spot in LST file.
Red line	Marks errors encountered. Cursor is positioned at first red line. When clicked on cursor jumps to Source (GMS) file location of error. Control click takes you to error location in LST file. Error description text is in execution process (output) window and in LST file. You can use a little red icon in source to jump to error message in LST file or a green arrow in the LST to go to error location in source.
Green Line	Causes STUDIO to open a file that was created during the run like a GDX, PUT or REF file

Using GAMS STUDIO

Getting Started

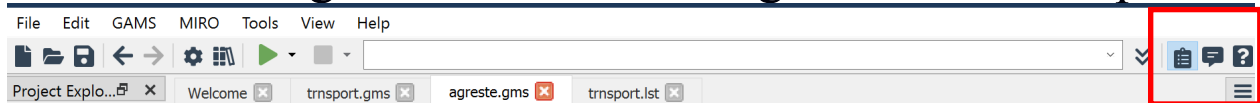
Navigate with process window

After double-clicking on any non-red line, or if we used the automatically open LST file option our main editing window is augmented by the **LST file** (see the tabs)



We can navigate as we would with an editor or word processor, as we are automatically in STUDIO text editor.

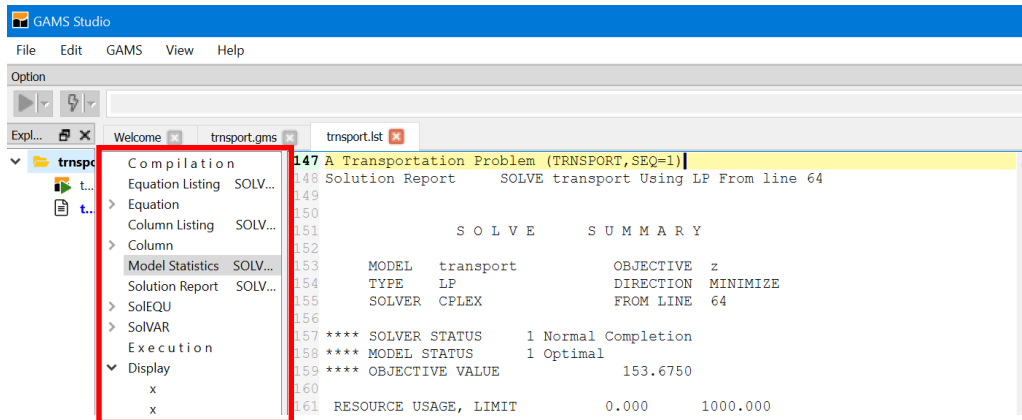
Sometimes I want to get the process window and explorer out of the way. I can do this through the view window and also through the **icons** on the right side of the top bar



Clicking on these opens and closes the explorer and process windows.

Using GAMS STUDIO Getting Started

Navigate with LXI window



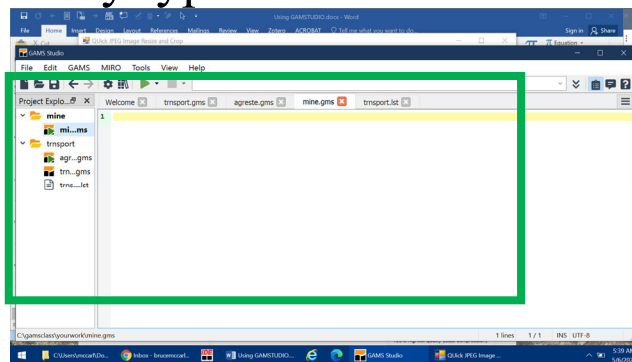
Navig. Line	Function and Destination When Clicked.
Compilation	Jumps to top of echo print in LST file
Error Mess.	Jumps to list of error messages when errors present
Equat. listing	Jumps to list of equation contents
Equation	Expandable allowing jump list of equation contents
Variab. listing	Jumps to list of variable
Variable	Expandable allowing jump to list of variable contents
Model stat.	Jumps to model statistics part of LST file
Solut.Report	Jumps to model summary solution report
SolEQU	Expandable allowing jump solution for each equation
SolVAR	Expandable allowing jump to solution for variable
Execution	Jumps to beginning of post solve execution
Display	Jump to displays of specific parameters and other items

Using GAMS STUDIO Working with your own file

Now you are ready to work with your own files. You may already have a file or you may need to create one.

Creating a new file - Two principal ways

1. Open an existing file through the library or from your hard disk. Then use the **save as** dialogue in **file** menu to change its name. Now modify contents. You may cut and paste as in other programs.
2. Open the **file** menu and use the **new** option. You will then be prompted for a name (I called it mine) with the extension **.gms** and then have a **blank screen** into which you may type GAMS instructions

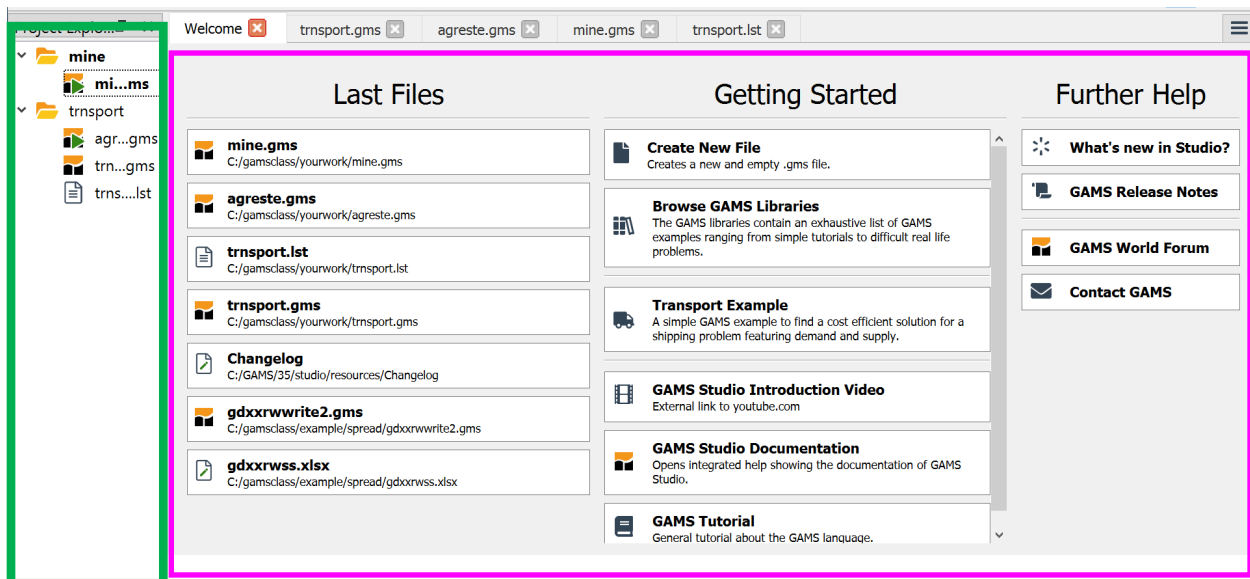


Save that file wherever you want and reopen it that will become the new default directory (**this only occurs upon an open not a save as**) and any subsequent non library opens, simple includes or saves will go there. Imports from the library go to the default workspace.

Using GAMS STUDIO

The Welcome and Explorer Windows

STUDIO can show a "Welcome" window (activated under the View menu) that remembers previous files plus gives access to documentation.



There as of now you get links to files recently used, and links to some documentation.

Also on the left hand side is an "Explorer" window that shows recent files and allows you to open by double clicking. This remains whether you are in the welcome page or editing files unless it is not checked in the View dialogue.

Using GAMS STUDIO Fixing Compilation Errors (tranerr.gms)

No one is perfect, errors always occur. STUDIO can help you in finding and fixing those errors.

Let's use a file from the class library `tranerr.gms` to illustrate. A run yields the windows below where right part is `process window` positioned to show first error and the center is the `lst file`

The screenshot shows the GAMS STUDIO interface with several windows open. The 'Process Log' window on the right displays the following error messages:

```
LimCol 0
IDE 1
LstTitleLeftAligned 1
Licensee: Small MUD - 5 User License
Texas A&M University, Agricul
C:\Users\mccarl\Documents\GAM
GAMS 35.1.0 Copyright (C) 1987-2021 G
--- Starting compilation
-- tranerr.gms (21) 3 Mb 1 Error
** Error 120 in C:\gamsclass\yourwork\
Unknown identifier entered as set
*** Error 140 in C:\gamsclass\yourwork\
Unknown symbol
--- tranerr.gms (51) 3 Mb 5 Errors
*** Error 120 in C:\gamsclass\yourwork\
Unknown identifier entered as set
*** Error 340 in C:\gamsclass\yourwork\
A label/element with the same name
to quote a label/element referen
set i / a,b,c /; parameter x(i);
```

The center window shows the LST file with the following error messages:

```
**** $257
**** 257 Solve statement not checked because of previous
57
Description
Transport model with syntax errors
Notebook Section Using GAMSIDE
GAMS Points Compilation Errors
other
include
**** 8 ERROR(S) 0 WARNING(S)
COMPILATION TIME = 0.000 SECONDS 3 MB 35.
USER: Small MUD - 5 User License G2006
Texas A&M University, Agricultural Economics
License for teaching and research at degree granting
```

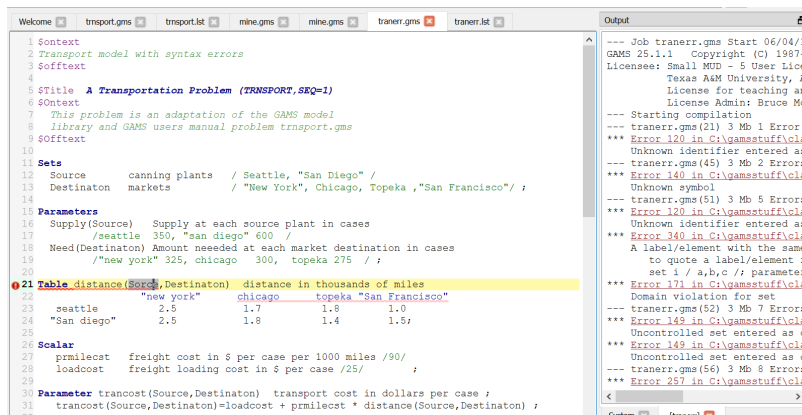
The **red lines** in the process window mark errors with error messages just below. To see where the errors occurred click on the very first red line which opens the source file or click on the black error message line just below the red error marking line to look at the LST file where the error appears and the exact placement will be marked with a \$.

Using GAMS STUDIO

Fixing Compilation Errors

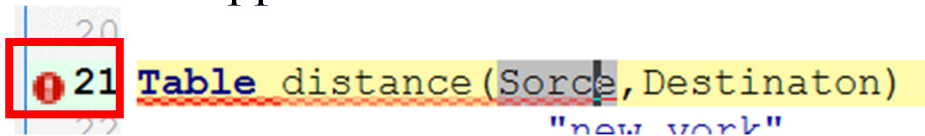
([tranerr.gms](#))

A click on the red line takes you to the place in the source where the error was made. The tip here is always start at the top of the process file when doing this and STUDIO now starts there.

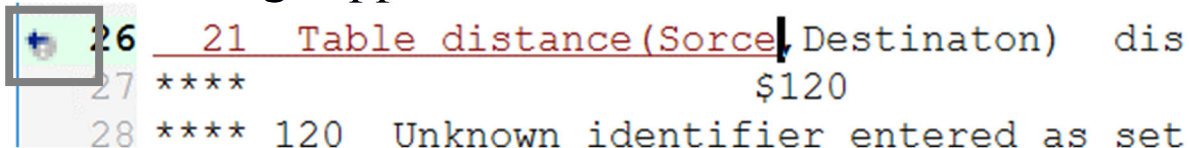


A double click on a black line takes you to the error location in the LST file.

Also in the source file window note the small red circle that appears on the line with the error



Clicking on that transfers you to the LST file where the error message appears.

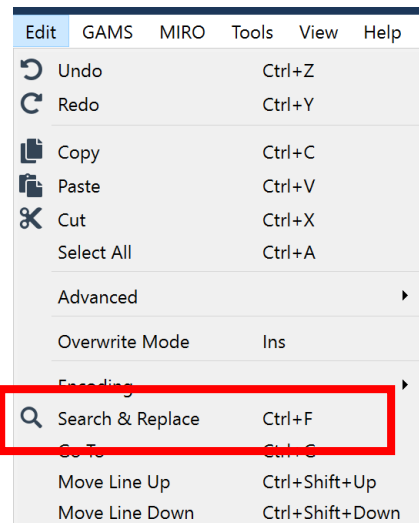


And there a small gray circle with an arrow in it that when clicked on will transfer you back to the source file

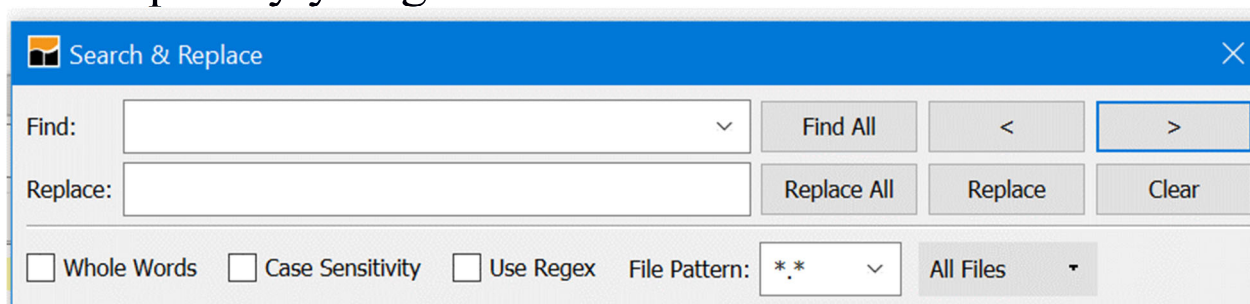
Using GAMS STUDIO

Ways to find and/or replace text strings

For finding you can use the search menu under the Edit tab or control F



Subsequently you get



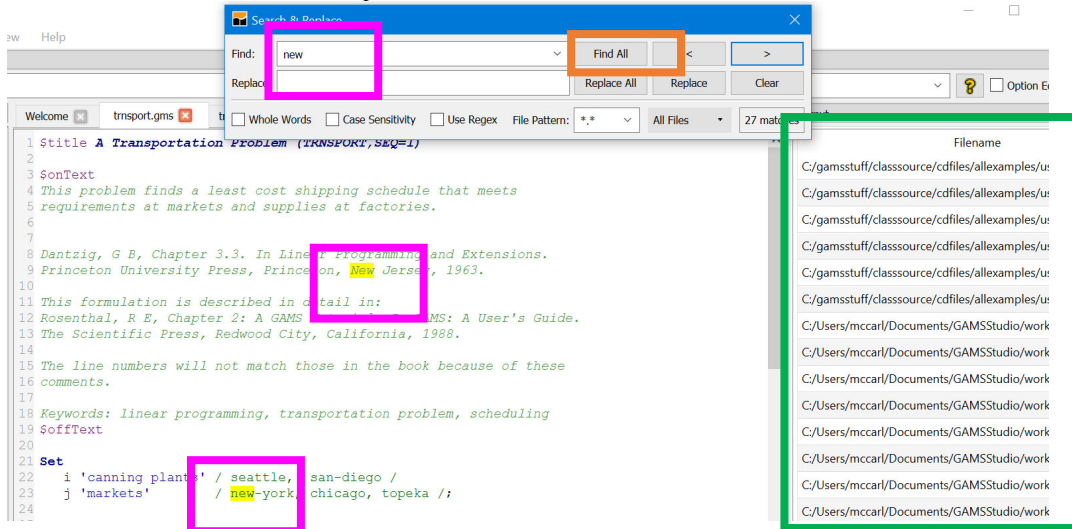
Which allows both search and replace. As of now this opens with the last option you used and may start with initially with an all files setting.

Note much to my dismay there is no find in files with a directory named and this is currently keeping me on the IDE as my main editor.

Using GAMS STUDIO

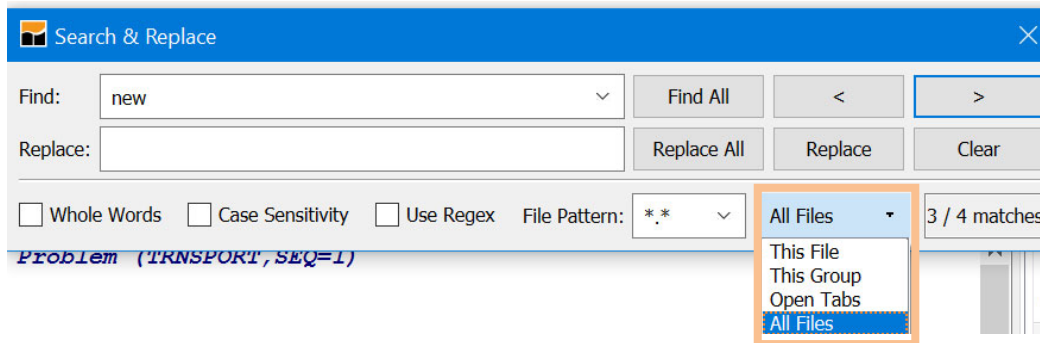
Ways to find and/or replace text strings

After the search you find



Where all if you select **find all** then all occurrences of the **search term** in the current file are highlighted and to the right there is a **list of all files that contain that search term** in all of the **files opened in the explorer**.

The search dialogue is straight forward with arrow keys to jump to next (F3 also works for forward and shift F3 for backward) and the **all files can be manipulated as shown**.



Using GAMS STUDIO Matching Parentheses (transport.gms)

The STUDIO provides you with a way of checking on how the **parentheses match up** in your GAMS code. This involves positioning your cursor just before an open parentheses or just after a close (Just after also works if there are not multiple parentheses). In either case the corresponding close or open is highlighted in **green**. Additionally positioning the cursor right before the beginning parentheses and tapping control b will lead you to the matching ending parentheses whether it be 1, 100, or 1000+ lines away and vice versa.

```
54 demand(j) satisfy demand at market j ;  
55  
56 cost..      z =e= sum((i,j), c(i,j)*x(i,j));  
57
```

When the expression in parentheses carries on the several lines the marking is in multiple lines and control b will transfer position back and forth

```
55  
56 cost..      z =e= (sum((i,j),  
57                  c(i,j)*x(i,j)  
58                  ));  
59
```

This feature will also match up { } or [] with a red marking if the types don't match.

Using GAMS STUDIO Moving Blocks

The STUDIO allows one to move text blocks through standard Cut and Paste Operations in two fashions.

1. One can identify a continuous block of text with the mouse or the keyboard (the latter involves putting the cursor at a beginning point then hold the shift key down and use the arrow keys).
2. One can identify a **column block of text** with the mouse or the keyboard (these involve holding **alt and shift down** then moving the mouse or the cursor with the arrow keys).

```

34
35 Table d(i,j) 'distance in thousands of miles'
36     new-york  chicago  topeka
37     seattle   2.5      1.7      1.8
38     san-diego 2.5      1.8      1.4;

```

In turn copy, cut, and paste can be done with the Edit menu or with control c, x and v respectively as in normal windows. Control insert also pastes.

Also if one types with a column block highlighted than the character goes in all rows of the block and replaces what was there before

```

29
30 b(j) 'demand at market j in cases'
31 / new-york 325
32 / chicago 300
33 / topeka 275 ;;
34

```

with an * typed becomes

```

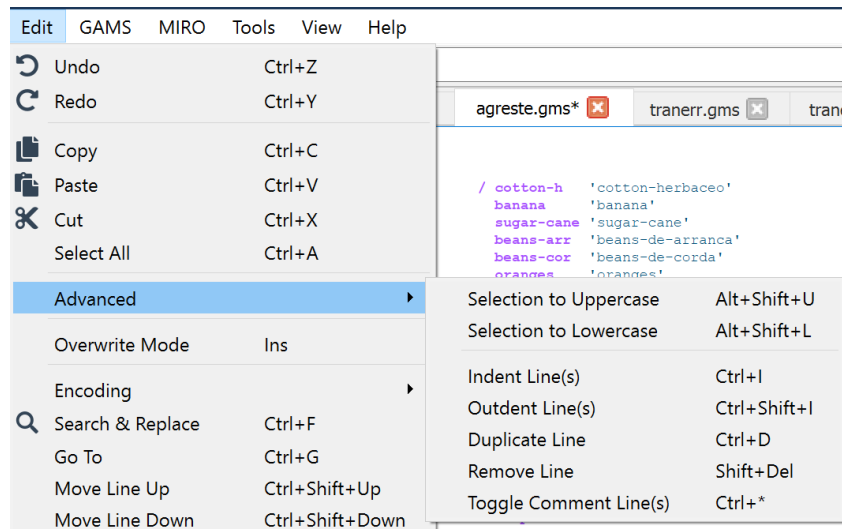
29
30 * b(j) 'demand at market j in cases'
31 * / new-york 325
32 * / chicago 300
33 * / topeka 275 ;;
34 *

```

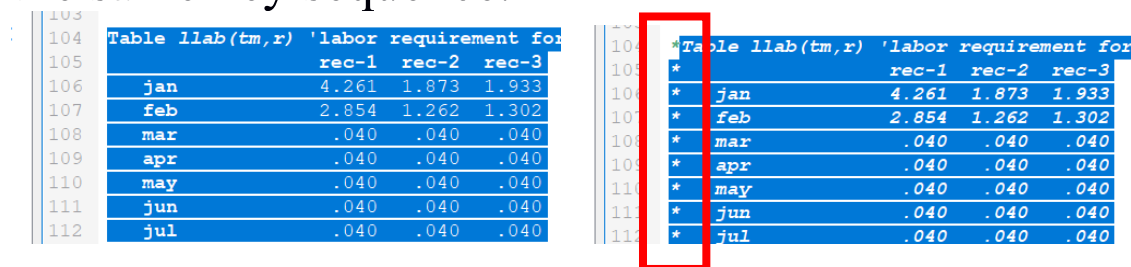
Using GAMS STUDIO

Other Editing Features

Under editing you get a few other choices that merit comment.

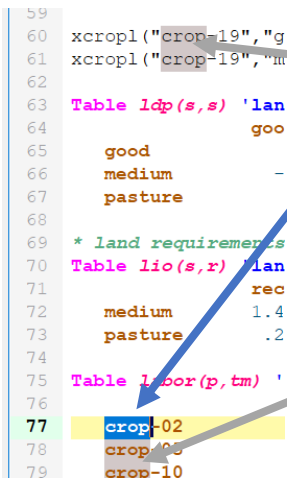


At the bottom of the edit you can see features to move lines up and down while under Advanced you can toggle case, add or remove indenting and convert a selected set of lines to or from comments. On the latter highlighting some code and typing control shift asterisk adds *** to every line in column 1**. Also you can reverse this or remove elsewhere by highlighting a block and pressing the same key sequence.



Using GAMS STUDIO Showing where a symbol appears

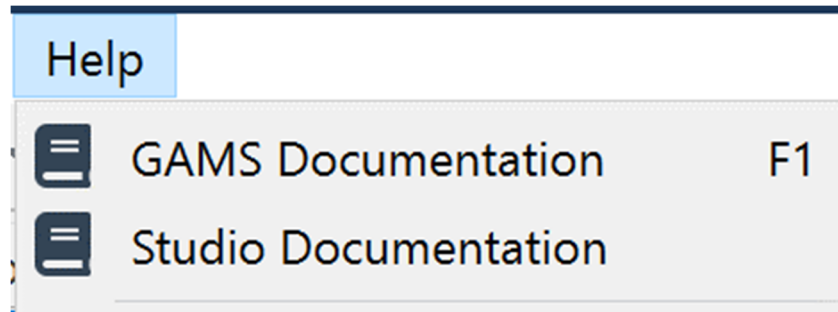
STUDIO has feature that identifies where a symbol is used. Taking the library file `agreste.gms` and positioning the cursor over a symbol like `"crop"` then double clicking places a gray highlight over all visible likely whole word instances of that symbol.



```
59
60 xcrop1("crop-19", "g
61 xcrop1("crop-19", "m
62
63 Table ldcp(s,s) 'lan
64         goo
65         good
66         medium -
67         pasture
68
69 * land requirements
70 Table lio(s,r) 'lan
71         rec
72         medium 1.4
73         pasture .2
74
75 Table labor(p,tm) '
76
77 crop-02
78 crop-05
79 crop-10
```

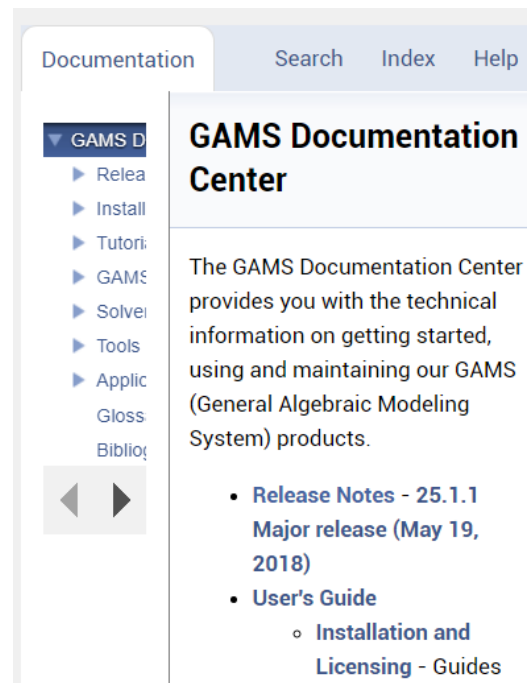
Using GAMS STUDIO -- Finding out more Accessing Documentation Via the Help

To access GAMS documentation choose help or F1 plus you can access documentation on STUDIO



And if you choose documentation you get access through a window in lower right corner as shown.

Therein you can navigate, search, and look at the index. Also if you get on the documentation page and go to the bottom you find the McCarl guide.

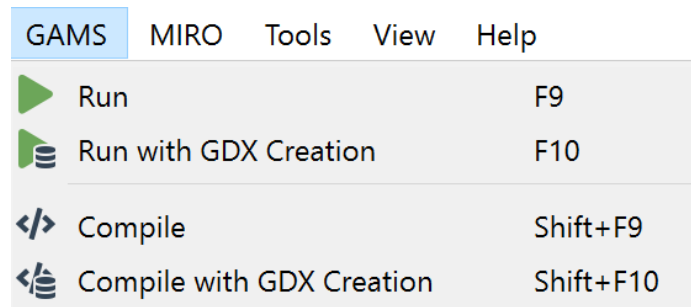


If your cursor is on a reserved word in the source file and you press F1 an index opens to all treatments of that word although in some strange places.

There is a STUDIO YouTube video that can be accessed through the Welcome page or in the general GAMS documentation under Tools.

Using GAMS STUDIO -- Finding out more Making GDX files

STUDIO can create and view GDX files. To create in STUDIO use the GAMS tab and choose a GDX creating choice. Alternatively use one of the other ways in the language as we will discuss elsewhere.



To browse a GDX file open it with the file open dialogue or click on green lines for ones created during a run in the process window and then for the transport example

The screenshot shows the GAMS STUDIO interface with a table of GDX items. The table has columns: Entry, Name, Type, Dim, Records, i, j, and Value. The 'i' and 'j' columns contain city names. The 'Value' column contains numerical values. The row for 'di' is highlighted in blue.

Entry	Name	Type	Dim	Records	i	j	Value
3	a	Parameter	1	2	capa	seattle new-york	2.5
4	b	Parameter	1	3	dema	seattle chicago	1.7
7	c	Parameter	2	0	trans	seattle	1.8
10	cost	Equation	0	1	defin	san-diego new-york	2.5
12	demand	Equation	1	0	satisf	san-diego chicago	1.8
5	di	Parameter	2	6	dista	san-diego topeka	1.4
6	f	Parameter	0	1	freigt		
1	i	Set	1	2	canni		
2	j	Set	1	3	mark		
11	supply	Equation	1	0	obse		
8	x	Variable	2	0	shipr		
9	z	Variable	0	1	total		

Here in the left-hand panel one gets an alphabetical listing of the items in the GDX file and their characteristics. Then clicking on a left-hand panel item leads to the item contents being revealed in the right hand panel. As of now you can reorder columns and filter values in table mode. With 3 plus dimensional items in table mode you can move things in rows into columns and vice versa


Using GAMS STUDIO -- Finding out more Examining GDX files

When you have opened a GDX file and selected an item then you get a window like that on the previous page

Entry	Name	Type	Dim	Records	Text	i	j	Value
3	a	Parameter	1	2	capacity of pla...	seattle	new-york	2.5
4	b	Parameter	1	3	demand at mar...	seattle	chicago	1.7
7	c	Parameter	2	0	transport cost i...	seattle	topeka	1.8
10	cost	Equation	0	1	define objectiv...	san-diego	new-york	2.5
12	demand	Equation	1	0	satisfy demand...	san-diego	chicago	1.8
5	di	Parameter	2	6	distance in tho...	san-diego	topeka	1.4

If in the right-hand panel I dragged the column labeled "i" to the right of value yielding.

j	Value	i
new-york	2.5	seattle
chicago	1.7	seattle
topeka	1.8	seattle
new-york	2.5	san-diego
chicago	1.8	san-diego
topeka	1.4	san-diego

Also by clicking on the header you cause the array to be sorted with a ^ marking the sort order and by clicking on  you can establish a filter that limits elements displayed

^ j	Value	i
chicago	1.7	seattle
chicago	1.8	san-diego
new-york	2.5	seattle
new-york	2.5	san-diego
topeka	1.8	seattle
topeka	1.4	san-diego

Filter

seattle

san-diego

Hide unselected items

Using GAMS STUDIO

A difficulty you will have

When using and teaching both the IDE and STUDIO, I find that file locations sometimes gives me fits. I have a rule of thumb to avoid problems, but you will most likely inadvertently not follow it.

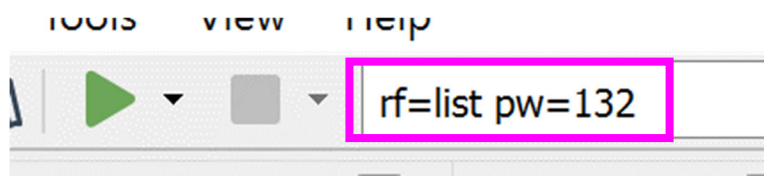
When working with a model with multiple files make sure they are located in the same directory location.

You do not have to follow this rule but deviations are the same as asking for trouble. When you are actively working on a file that you have opened from a directory then STUDIO will look for files in [the directory where the active file and group is located](#). I would like STUDIO to open all in a single group if I specified that but the library manager fails to do that.

Also when you run a file which contains includes GAMS will look in the [directory where the active file is located](#).

Using GAMS STUDIO Command Line Parameters

Experienced DOS or UNIX based GAMS users are used to having command line parameters associated with their GAMS execution commands. In STUDIO a **command line parameter entry box** is available just to the right of the execute button the parameters are associated with a file and STUDIO will remember these whenever the file is opened.

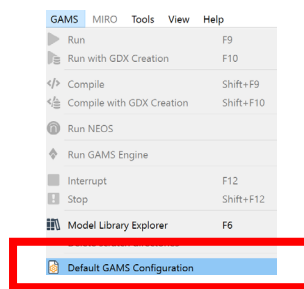


This is particularly useful for save, restart, limrow, limcol, pw and rf parameters as once they are defined they are associated with every subsequent use of the file.

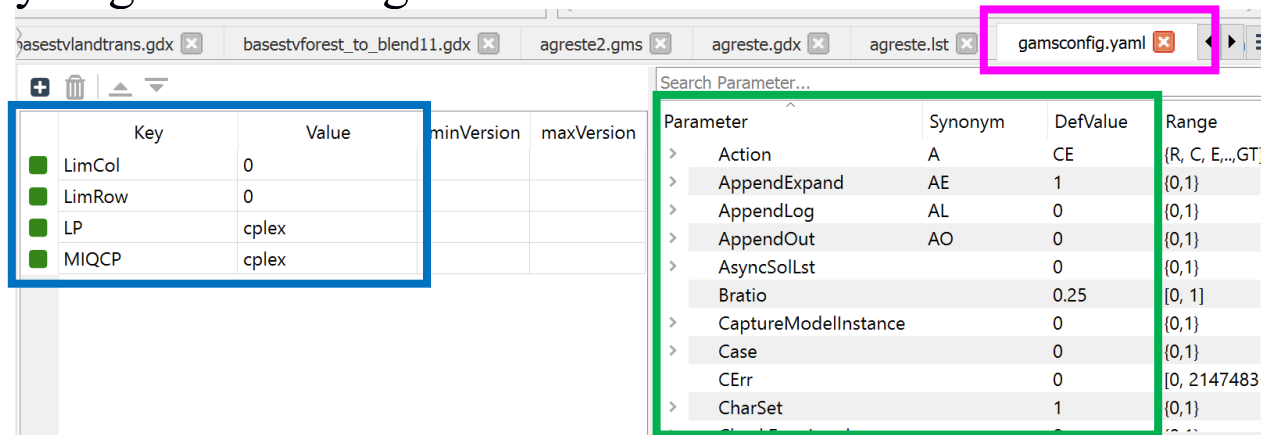
As of now you can also provide command line parameters for use in every run (like a wide page width) through the yaml file as discussed on the next page.

Using GAMS STUDIO Default Command Line Parameters – yaml

There are cases when you would want to supply parameters to all jobs that you run. This can be done by manipulating the yaml file. In particular if you go into the GAMS menu choice under **default configuration**




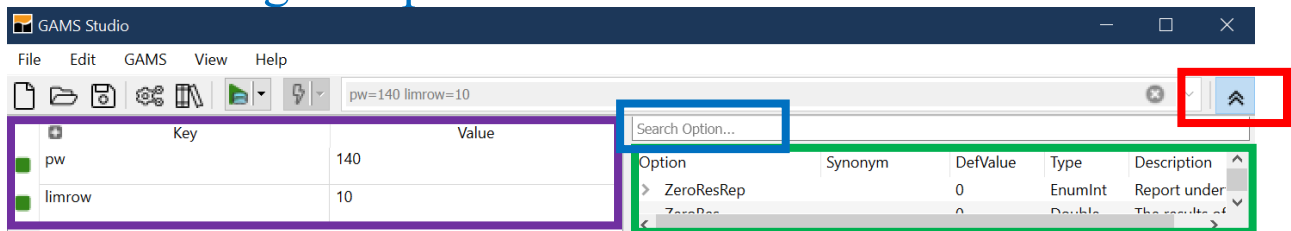
you get something like



Where the default values appear on the left and you can select parameters to provide default values for in the right box. Note these choices are overridden if you enter things in the command line box as on the previous page.

Using GAMS STUDIO Command Line Parameters

GAMS provides assistance with command line parameters through the command line parameter editor. To activate it check the  symbol at the far right up top. Then part of the screen becomes as below. If the left hand part existing command line parameters in use are listed along with their values. On the right all possible command line parameters are listed with definitions and a search dialogue is provided.



The search dialogue looks for text in any of the fields, thus typing in **wid** brings up page width as does **pw** and **maxi** brings up those with maximum in the description.

This is particularly useful for save and restart and a number of other parameters like limrow, limcol, and pw parameters as once they are defined they are associated with every subsequent use of the file.

Note pressing F1 while the cursor is on a command line entry opens help to a place with a detailed description of that parameter.

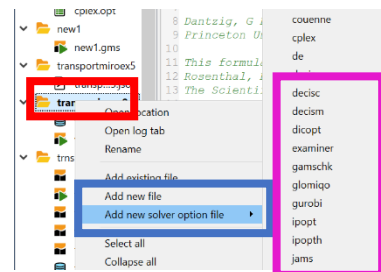
Using GAMS STUDIO Solver option files

STUDIO contains allows you to edit or create a solver option file. When using the solver **conopt** you create **conopt.opt** or with **cplex** use **cplex.opt** (or **.op1** or **.o10** or **.111**) and more generally **solvername.opt**

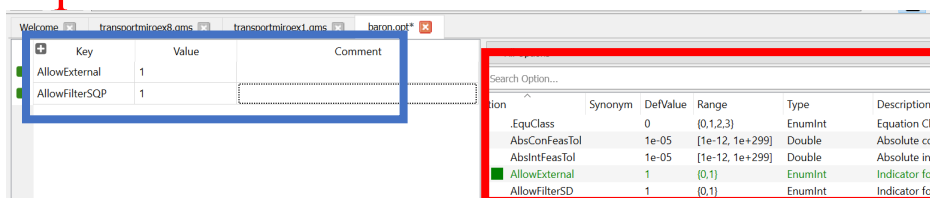
Specifying such a file allows you to manipulate the way the solver works on the model. For example, you can alter scaling, tolerances, iteration limits, solver algorithm, and branch and bound strategy along with many other things.

In STUDIO the option file editor is accessed by either

- Right clicking on the **group name** for a group where you want an option file then choosing **add new solver option file** and picking the **appropriate solver name**



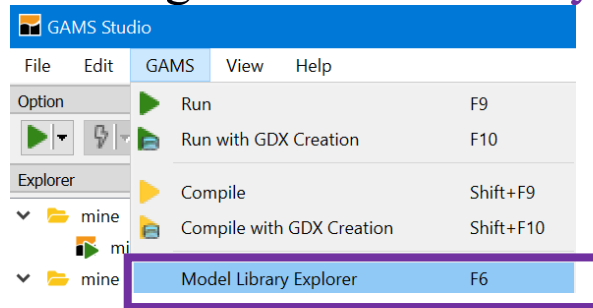
- Creating a file with a name like **conopt.opt** then when you bring that up in the editor the option file editor appears. In the editor you see **options in use**, **possibilities and definitions**



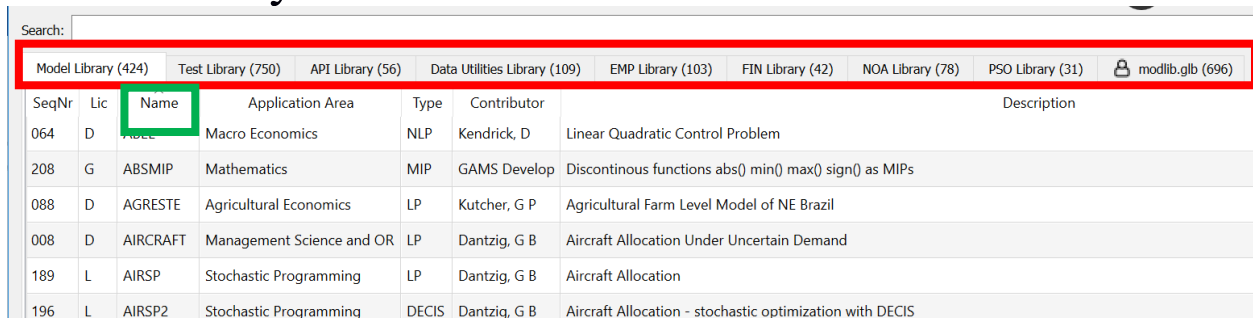
//

Using GAMS STUDIO Using Libraries

STUDIO allows one to access libraries under the GAMS menu item and choosing the **Model Library Explorer**



in turn this opens a screen which identifies names of the libraries in tabs with a count of included files and, for the tab selected, **the library contents**. Furthermore across the top are tabs that when clicked on cause sorts. The picture below came after I clicked on the tab name with the model library active

A screenshot of the Model Library Explorer interface. At the top, there is a search bar and a row of tabs for different libraries: 'Model Library (424)', 'Test Library (750)', 'API Library (56)', 'Data Utilities Library (109)', 'EMP Library (103)', 'FIN Library (42)', 'NOA Library (78)', 'PSO Library (31)', and 'modlib.glb (696)'. The 'Model Library (424)' tab is selected. Below the tabs is a table with the following columns: 'SeqNr', 'Lic', 'Name', 'Application Area', 'Type', 'Contributor', and 'Description'. The 'Name' column is highlighted with a green box.

SeqNr	Lic	Name	Application Area	Type	Contributor	Description
064	D	ABCE	Macro Economics	NLP	Kendrick, D	Linear Quadratic Control Problem
208	G	ABSMIP	Mathematics	MIP	GAMS Develop	Discontinuous functions abs() min() max() sign() as MIPs
088	D	AGRESTE	Agricultural Economics	LP	Kutcher, G P	Agricultural Farm Level Model of NE Brazil
008	D	AIRCRAFT	Management Science and OR	LP	Dantzig, G B	Aircraft Allocation Under Uncertain Demand
189	L	AIRSP	Stochastic Programming	LP	Dantzig, G B	Aircraft Allocation
196	L	AIRSP2	Stochastic Programming	DECIS	Dantzig, G B	Aircraft Allocation - stochastic optimization with DECIS

Users can define their own libraries. They have to define what's called a GLB file and the formatting creation of this is discussed in my newsletter in the second issue.

<https://www.gams.com/mccarl/newsletter/news2.htm>.

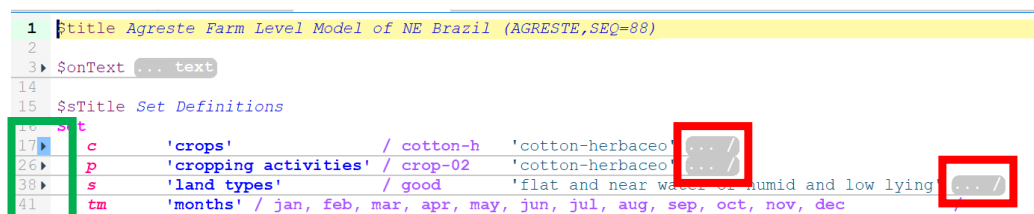
Using GAMS STUDIO Folding Code

STUDIO allows one mark or otherwise select code to hide when in the editor in a procedure called folding. This does not remove that code from the GMS file but just hides it from view in the STUDIO editor.

Several kinds of code are folded including

- code between matching parentheses across multiple lines
- code between \$ontext and \$offtext, \$onfold and \$offfold, \$onput and \$offput, \$ifthen and \$endif and others
- Code in multiple line definitions of sets, items surrounded by parentheses, things in multiple line loops, ifs and others.

Markers are inserted to indicate folded code. One to the **right of the line numbers** and another in the **first line of the folded block**.



```
1 $title Agreste Farm Level Model of NE Brazil (AGRESTE,SEQ=88)
2
3 $onText ... text
14
15 $sTitle Set Definitions
16
17 c 'crops' / cotton-h 'cotton-herbaceo
26 p 'cropping activities' / crop-02 'cotton-herbaceo
38 s 'land types' / good 'flat and near water
41 tm 'months' / jan, feb, mar, apr, may, jun, jul, aug, sep, oct, nov, dec
```

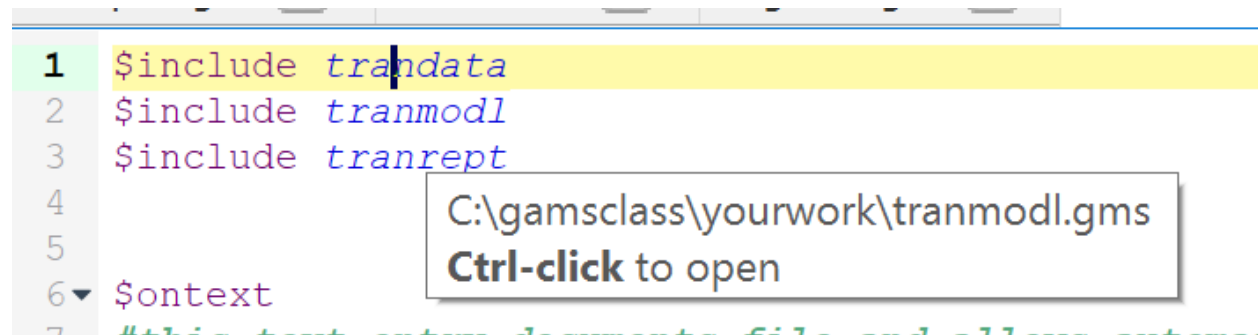
To toggle folding state of a section either the **fold-marker by the line numbers** can be clicked or the shortcut key **Alt - L** when cursor is on a foldable line. All folds are activated or deactivated using **View menu or Alt - O** and **Alt - Shift - O**.

Using GAMS STUDIO

Bookmarks and accessing include files

STUDIO allows you to add bookmarks in the code then access them under view and bookmarks. Unfortunately they are not saved from one use of STUDIO to another and thus disappear when STUDIO is closed. Hopefully GAMS will create a way to store them like it does command line parameters.

Studio also allows you to automatically open files referenced by include statements as long as they are located in your STUDIO working directory. To do this you hover over a file name in an include statement and then type control click as prompted



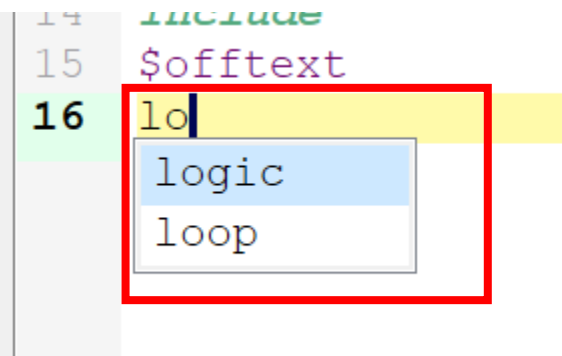
```
1 $include trandata
2 $include tranmodl
3 $include tranrept
4
5
6 $ontext
7 #this text enters documents file and it enters
```

C:\gamsclass\yourwork\tranmodl.gms
Ctrl-click to open

Using GAMS STUDIO Code completion

STUDIO offers completion for keywords known by the compiler.

The code completer is active by default and shows keywords that matches the current syntax and starts with the already typed characters of the word to the left of the cursor. Below I started typing **lo** and the **completer showed up with choices**

A screenshot of the GAMS STUDIO code editor. The editor shows three lines of code: line 14 with the keyword 'include', line 15 with '\$offset', and line 16 with 'lo'. The text 'lo' on line 16 is highlighted in yellow. A red rectangular box is drawn around the text 'lo' and the completion list. The completion list is a white box with a blue border containing two options: 'logic' and 'loop'. The 'logic' option is highlighted with a blue background. The line numbers 14, 15, and 16 are visible on the left side of the editor.

```
14 include
15 $offset
16 lo
```

- logic
- loop

The selected word can be inserted with the keys tab or enter/return or by double clicking a word in the list.

Using GAMS STUDIO

Using REF files Listing and Unraveling Data items

There is a procedure in STUDIO called to display GAMS reference files that gives information on where things are defined and used.

Why? GAMS Modelers sometimes have to deal with complex implementations that

- Use include statements to incorporate numerous files.
- Have been developed by others
- Have a complex structure with definitions and uses of items widely spread in a file or files.
- Contain items that are defined but never used.
- Were developed some time ago but are not extensively documented.

When faced with such cases one often asks

- Are there items defined in the program that are not used and if so what are they and where are they.
- In what files is an item defined, declared and used.

To resolve these questions some years ago I wrote a Fortran program called GAMSMAP but never made it widely available. Subsequently its function was included into the GAMSIDE and now in STUDIO.

Using GAMS STUDIO

Using REF files - Listing and Unraveling Data items

When a Ref file is created for a run (using a command line entry like rf=list) and then opened by **clicking on its line in the LOG file** (this is located in the LOG just at the end of compile step toward the top)

```

Process Log
C:\Users\mccarl\Documents\GAMS\ga
GAMS 35.1.0 Copyright (C) 1987-2021 GAMS
--- Starting compilation
--- agreste.gms (298) 3 Mb
--- RefFile C:\gamsclass\yourwork\list.ref
--- Starting execution elapsed 0:00:00.019
--- agreste.gms (292) 4 Mb
--- Generating LP model agreste
--- agreste.gms (294) 4 Mb

```

or simply opening it STUDIO creates a window as follows

Entry	Name	Type	Dim	Domain	Location	Line	Column
15	abs	Function	0				
392	ADJUST	Parameter	4	(ALLI,subreg,C			
234	AGPRODUCT	Set	1				
236	AGPRODUCT	Parameter	4	(REGIONS,ALI			
242	ALLCOM	Set	1	(ALLI)			
138	ALLI	Set	1				
152	ANIMAL	Set	1				
325	ARTIF	Variable	1	(ALLI)			
353	ARTIFICIAL	Equation	1	(CROP)			
327	ARTS	Variable	1	(ALLI)			
384	ascale	Parameter	1				

Using GAMS STUDIO

Using REF files - Listing and Unraveling Data items

Using `rf=list` in the GAMS library file `agreste` we get

Entry	Name	Type	Dim	Domain	Location	Line	Column
15	abs	Function	0				
392	ADJUST	Parameter	4	(ALL,subreg,C			
234	AGPRODUCT	Set	1				
236	AGPRODUCT	Parameter	4	(REGIONS,ALI			
242	ALLCOM	Set	1	(ALL)			
138	ALLI	Set	1				
152	ANIMAL	Set	1				
325	ARTIF	Variable	1	(ALL)			
353	ARTIFICIAL	Equation	1	(CROP)			
327	ARTS	Variable	1	(ALL)			
384	ascale	Parameter	1				

The **column on the left** gives tags that can be clicked on and the key ones are

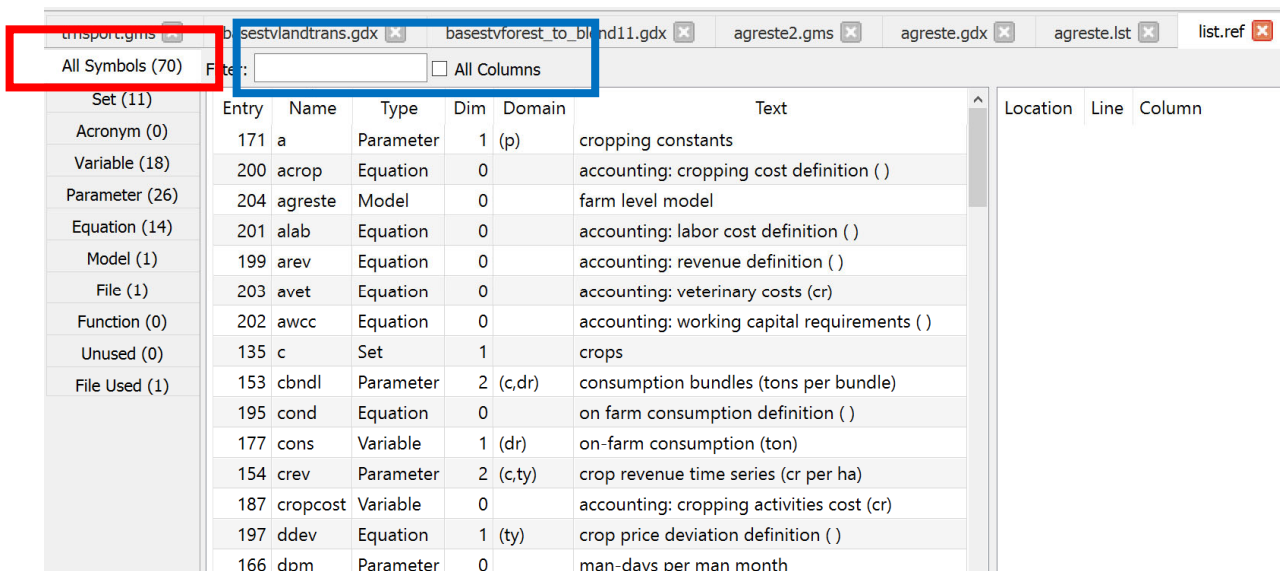
- All Symbols** lists all items declared and a count of them
- Set** lists all sets in the program and a count
- Variable** Lists all variables and a count
- Parameter** Lists all parameters (items defined in scalar, parameter or table statements) and a count.
- Equation** Lists all equations in the program and the names of files in which they appear
- Model** Lists all models that exist in the program and the names of files in which they appear
- Unused** Lists all items that are declared or that are not used anywhere in the model
- File used** Lists almost all files included into the model (note ones without GAMS keywords like included data files are not listed)

Using STUDIO

Listing and Unraveling Data items

All Symbols Tab

Choosing **all symbols row** causes the output to appear as follows (for the model library `agreste.gms` example)



	Entry	Name	Type	Dim	Domain	Text	Location	Line	Column
Set (11)									
Acronym (0)	171	a	Parameter	1	(p)	cropping constants			
Variable (18)	200	acrop	Equation	0		accounting: cropping cost definition ()			
Parameter (26)	204	agreste	Model	0		farm level model			
Equation (14)	201	alab	Equation	0		accounting: labor cost definition ()			
Model (1)	199	arev	Equation	0		accounting: revenue definition ()			
File (1)	203	avet	Equation	0		accounting: veterinary costs (cr)			
Function (0)	202	awcc	Equation	0		accounting: working capital requirements ()			
Unused (0)	135	c	Set	1		crops			
File Used (1)	153	cbndl	Parameter	2	(c,dr)	consumption bundles (tons per bundle)			
	195	cond	Equation	0		on farm consumption definition ()			
	177	cons	Variable	1	(dr)	on-farm consumption (ton)			
	154	crev	Parameter	2	(c,ty)	crop revenue time series (cr per ha)			
	187	cropcost	Variable	0		accounting: cropping activities cost (cr)			
	197	ddev	Equation	1	(ty)	crop price deviation definition ()			
	166	dpm	Parameter	0		man-days per man month			

This shows all symbols used. The columns give the symbol name (Name), its type (Type) – set parameter etc, its dimension (Dim) – # of sets it is defined over, names of sets that dimension it (DOMAIN) the explanatory text used in its declaration (TEXT).

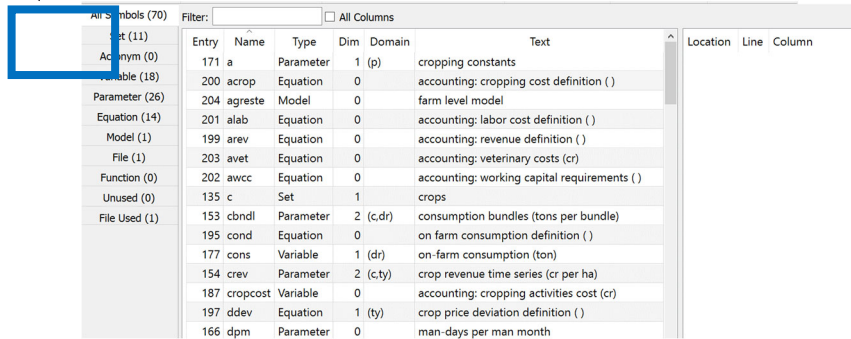
Things can be sorted by clicking on the column headings.

One can search for symbols using the **filter box**

Using STUDIO

Listing and Unraveling Data items Sets, Parameters etc. Tabs

The display contains 5 lines that when activated give information for sets, parameters etc. The resultant display lists all items falling in a class (for all things that are sets or parameters etc.) and information on their characteristics and use. Specifically one gets output as follows (for the sets in this case)



Once **set is clicked on** the entries tell the names of the files in which certain things happen relative to the identified items.

Entry	Name	Dim	Domain	Text	Location	Line	Column
135	c	1		crops	<ul style="list-style-type: none"> (1) Declared in <ul style="list-style-type: none"> agreste.gms 45 6 		
142	dr	1		family consumption bundle alternatives			
143	km	1		technology characteristics	<ul style="list-style-type: none"> (1) Defined in <ul style="list-style-type: none"> agreste.gms 45 53 		
136	p	1		cropping activities	<ul style="list-style-type: none"> (0) Assigned (0) Implicitly Assigned 		
140	r	1		livestock feeding alternatives	<ul style="list-style-type: none"> (3) Controlled in <ul style="list-style-type: none"> agreste.gms 264 34 agreste.gms 272 19 agreste.gms 286 45 		
137	s	1		land types			
139	sc	1	(s)	crop lands			
145	sp	1		Aliased with s			
138	tm	1		months	<ul style="list-style-type: none"> (6) Referenced in <ul style="list-style-type: none"> agreste.gms 119 17 		
141	tv	1		year			

Clicking on a column name causes it to be sorted.

Using STUDIO

Listing and Unraveling Data items Sets, Parameters etc. Tabs

Entry	Name	Dim	Domain	Text	Location	Line	Column
135	c	1		crops			
142	dr	1		family consumption bundle alternatives	agreste.gms	45	6
143	km	1		technology characteristics			
136	p	1		cropping activities	agreste.gms	45	53
144	ps	2	(p,s)	process-land possibilities			
140	r	1		livestock feeding alternatives			
137	s	1		land types			
139	sc	1	(s)	crop lands	agreste.gms	264	34
145	sp	1		Aliased with s	agreste.gms	272	19
138	tm	1		months	agreste.gms	286	45
141	tv	1		vear	agreste.gms	119	17

The categories include the file name, line and column where items are declared, defined and used.

Declared places where the named item is declared in a Set, Parameter, Table, Scalar, Variable, Equation, Acronym, File or Model command. This will be the first appearance.

Defined places set elements or data are explicitly entered. For equations this tells where the .. specification begins.

Assigned places where items appear on left hand side of an assignment statement

Implicitly-Assigned places where an equation or variable has data put into it by the results of a solve statement.

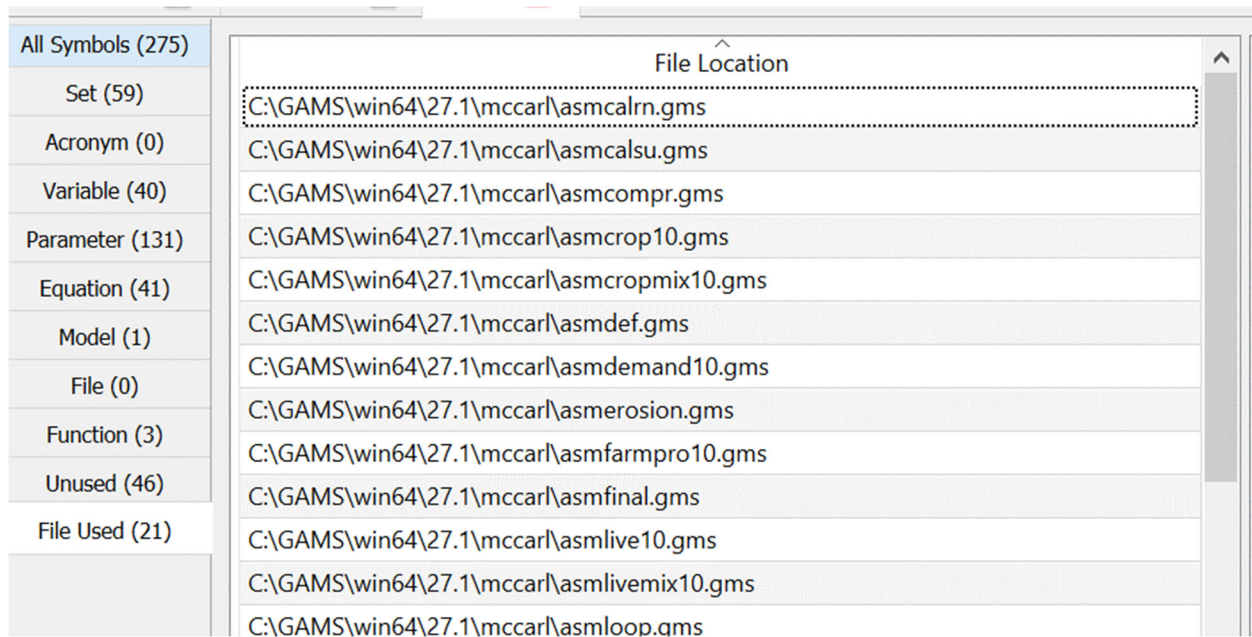
Controlled places where set is used in controlling a sum or defining an equation

Ref places where item is on left hand side of assignment statement or in a model equation

Clicking on lines here opens the subject file in the specific place

Using STUDIO Listing and Unraveling Data items File used Tab

The file used tab causes the output to appear as follows
(for the asmsmall10.gms example)



	File Location
All Symbols (275)	
Set (59)	C:\GAMS\win64\27.1\mccar\asmcalrn.gms
Acronym (0)	C:\GAMS\win64\27.1\mccar\asmcalsu.gms
Variable (40)	C:\GAMS\win64\27.1\mccar\asmcompr.gms
Parameter (131)	C:\GAMS\win64\27.1\mccar\asmcrop10.gms
Equation (41)	C:\GAMS\win64\27.1\mccar\asmcropmix10.gms
Model (1)	C:\GAMS\win64\27.1\mccar\asmdef.gms
File (0)	C:\GAMS\win64\27.1\mccar\asmdemand10.gms
Function (3)	C:\GAMS\win64\27.1\mccar\asmerosion.gms
Unused (46)	C:\GAMS\win64\27.1\mccar\asmfarmpro10.gms
File Used (21)	C:\GAMS\win64\27.1\mccar\asmfinal.gms
	C:\GAMS\win64\27.1\mccar\asmlive10.gms
	C:\GAMS\win64\27.1\mccar\asmlivemix10.gms
	C:\GAMS\win64\27.1\mccar\asmloop.gms

This gives the names of the files included in the program
with their full path references.

Using STUDIO Listing and Unraveling Data items Unused Tab

The unused tab identifies items that are declared (in set parameter etc statements) but are never used on the right hand side of an assignment (=) statement or in a model equation.

The output for the model `asmall10.gms` (which is distributed with the McCarl user guide) is as follows

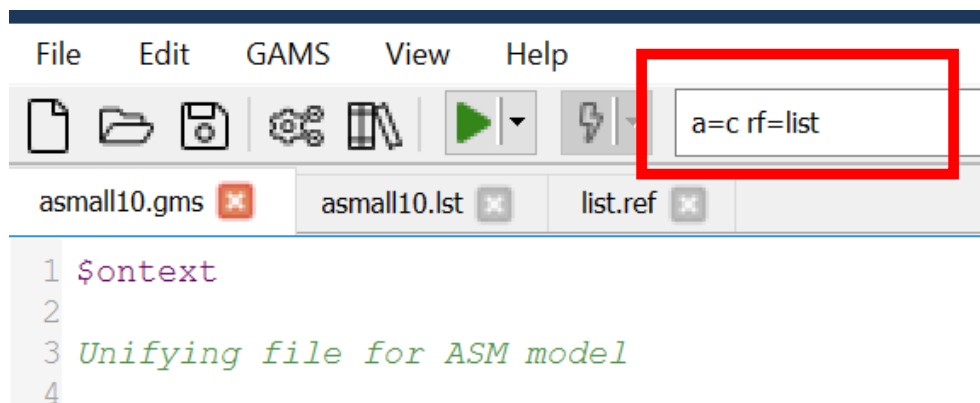
Entry	Name	Type	Dim	Domain	Text	Location	Line	Column
236	AGPRODUCT	Parameter	4	(REGIONS,ALLI...				
219	AUMSSUM	Parameter	2	(REGIONS,AUM...	AUMS USE SUMMARY			
202	BASEOUTPUT	Parameter	0					
261	commodc	Parameter	4	(REGIONS,type...				
240	CROPREG	Parameter	2	(REGIONS,CROP)	REGIONAL CROP PRODUCTION REPORT			
198	EROSION	Parameter	3	(REGIONS,CRO...	EROSION DATA			
249	EROSIONd	Parameter	2	(REGIONS)	EROSION			
266	fprevbal	Parameter	0					
229	FWELFARE	Parameter	2	(ALLI,ITEMFOR)	FOREIGN WELFARE			
270	grossrv	Parameter	2	(RFGIONS.PRI...				

Location	Line	Column
(1) Declared in		
asm surep.gms	126	20
(0) Defined		
(0) Assigned		
(0) Implicitly Assigned		
(0) Controlled		
(0) Referenced		

which shows the same information as in the All Symbols tab except when an item is clicked on one gets information on the files where it is declared

Using STUDIO Listing and Unraveling Data items Steps to Using Refreader

This will only work after a particular “reference file” has been created by a GAMS run. The file is generated by adding the rf option to the command line call entering a command in the command line box of STUDIO as follows
(a=c causes compilation only)



Once the file has been run with the rf option the logfile is augmented with the green line identifying the ref file name as below

```
Output
--- . asmloop.gms (171) 4 Mb
--- asmall10.gms (24) 4 Mb
--- . asmfinal.gms (52) 4 Mb
--- asmall10.gms (34) 4 Mb
--- RefFile C:\GAMS\win64\27.1\mccarl\list.ref
*** Status: Normal completion
--- Job asmall10.gms Stop 06/11/19 19:28:24 elap
```

Double clicking on this opens the reference file window.

Using STUDIO

Listing and Unraveling Data items

The run of GAMS with the rf option places the name of the ref file in green in the GAMS log file and double clicking on that line causes the refreader program to run and create its output.

Several notes about the run

- The rf= command specifies the name of the file that STUDIO needs to open. STUDIO expects it to have the extension .ref. Typically I use rf=list that creates the file list.ref.
- The rf file only covers the program components in a run and does not include any information from restart files. In general to get the most complete report it is best to explicitly use all the files in one program without use of save and restart
- It is often useful to just generate the reference file without any execution on behalf of the GAMS program. This is done by including the a=c option on the command line or in the command parameter box.

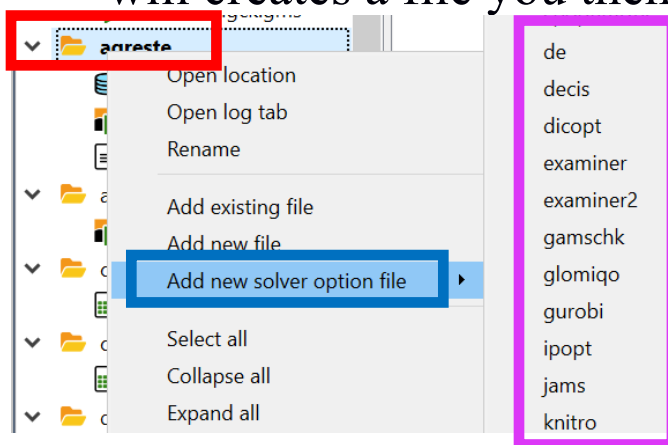
Using GAMS STUDIO Editing Solver Option Files

Sometimes it is desirable to change solver specific options. This includes modifying things like branch and bound strategy, degree of scaling, barrier versus simplex solution approach, solution tolerances etc. The specific options allowed are described in the solver manuals.

STUDIO contains a solver option file editor that presents a solver specific menu of all possible options, default values and ones that have been altered. This is activated by either

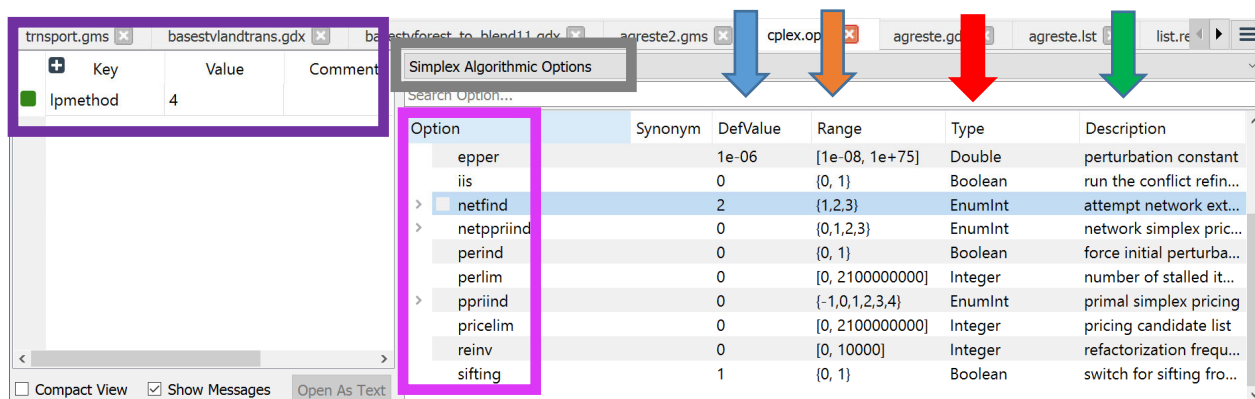
Opening or saving a file with solvername.opt (Like MINOS.opt)

Right clicking on a **group file name** in the STUDIO Explorer then selecting **add new solver option file** and subsequently selecting one of the many available **solver names** (only a few shown). This will create a file you then need to save.



Using GAMS STUDIO Editing Solver Option Files

Once it opens the general format of the solve option editor is as follows. Note the specific content depends on the solver chosen and in this case I use GAMSCHK.



Here in the right part we see a list of **all possible options** for a selected class of options for this solver. A brief description of **default values**, **entry type**, if relevant **allowable settings** and **a one sentence description of the option**. In the left part we see the **current entries** in the option file and **their settings**.

Subsequently when this file is opened or its tab activated then this solver option editor appears.

Note pressing F1 while on an option opens help to a place with a detailed description of that option

Using GAMS STUDIO When is it Not Worth Using?

There are costs and benefits of these approaches.

The STUDIO is much easier for simple models and can run across platforms.

The DOS/command line approach is generally better for models in customized environments

A **development strategy** for more complex implementations

- A. Use the STUDIO to get it right
- B. Debug components of large models using save and restart
- C. Then use script files or DOS/UNIX with batch files such as
GAMS mymodel -lo 0 -s ./t/save1
call myprogram.exe
GAMS moremod -lo 0 -r ./t/save1

Using GAMS STUDIO

What does it not do?

A number of IDE features are currently missing. Just to inventory (note the last couple are pretty unimportant to me.)

- Find in files with directory to look specified
- The ability to do tiled windows and look at multiple cases of a file
- Ability to save output from the reference file viewer
- Ability to change license file
- Ability to change the file associations when clicking on them in Windows Explorer (you get more than I like at the moment and sometimes I want more)
- Project files (mixed blessing)
- Text file differencing (can use other programs)
- Spell checking

Using GAMS STUDIO

What does it not do so well?

Here are also some difficulties with it as of today

- Washed out colors – on my machine the syntax and error coloring is not very prominent.
- Unusual bolding and choice of items to color in the syntax coloring with for example the column names in the table bolded but not the row names.
- Confusing wording of just what is there for user defined library choice under settings and miscellaneous. Right now it simply tells you the name of the directory where library files are kept but when I first used it I thought it was where you would tell STUDIO about a new library and where it is located.
- Excessive library choices and no way to put the ones you have no interest in somewhere out of the way. Plus any user defined libraries are at the defined and there is no way to reorder.