Recent Enhancements in GAMS

Lutz Westermann, GAMS Software GmbH
 Agenda

What is GAMS?

Enhancements in GAMS

Enhancements around GAMS

Summary
Company Background

Roots: World Bank, 1976

GAMS Development Corporation (Washington)

Tool Provider: General Algebraic Modeling System

Went commercial in 1987

GAMS Software GmbH (Cologne, Braunschweig) 1996
Algebraic Modeling Languages (AML)

1. High-level computer programming languages
   - Formulation of mathematical optimization problems
   - Notation similar to algebraic notation

2. Do not solve problems directly, but offer links to state-of-the-art algorithms ("solver-links")

Source: http://en.wikipedia.org/wiki/Algebraic_modeling_language
What does a modeler have to think about?

1. Problem
2. Mathematics
3. Programming
4. Performance
5. Scalability
6. Connectivity
7. Deployment
8. Maintenance (Life Cycle)
9. ...

Why is GAMS a tool for him?
Broad User Community and Network

GAMS used in more than 120 countries

25+ Years GAMS Development
General Algebraic Modeling System

Broad User Community and Network

More than 10,000 licenses

50% academic users, 50% commercial users

6,000+ monthly downloads of the free system

25+ Years GAMS Development
## Broad Range of Application Areas

<table>
<thead>
<tr>
<th>Agricultural Economics</th>
<th>Applied General Equilibrium</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chemical Engineering</td>
<td>Economic Development</td>
</tr>
<tr>
<td>Econometrics</td>
<td>Energy</td>
</tr>
<tr>
<td>Environmental Economics</td>
<td>Engineering</td>
</tr>
<tr>
<td>Finance</td>
<td>Forestry</td>
</tr>
<tr>
<td>International Trade</td>
<td>Logistics</td>
</tr>
<tr>
<td>Macro Economics</td>
<td>Military</td>
</tr>
<tr>
<td>Management Science/OR</td>
<td>Mathematics</td>
</tr>
<tr>
<td>Micro Economics</td>
<td>Physics</td>
</tr>
</tbody>
</table>

25+ Years
GAMS Development
Strong Development Environment

GAMS IDE

- Project management
- Editor / Syntax coloring / Spell checks
- Listing file / Tree view / Syntax-error navigation
- Model Debugging / Profiling
- Solver selection / Option selection
- Data viewer (GDX)
  - Export
  - Charting
- GAMS Processes Control
- Model Libraries
Free Model Libraries

- GAMS Model Library
- GAMS Test Library
- GAMS Data Utilities Models
- GAMS EMP Library
- Practical Financial Optimization Models

More than 1250 models!
Design **Principles**

1. Simple modeling language with a balanced mix of declarative and procedural elements

2. Open architecture and interfaces to other systems, independent layers
Simple Declarative Language

1. Few basic language elements: sets, parameters, variables, equations, models
2. Language similar to mathematical notation
3. Easy to learn
4. Model is executable description of the problem
5. Lot’s of code optimization under the hood
Mix of Declarative and **Procedural Elements**

Procedural elements like loops, for, if, macros and functions

- Allow to build complex problem algorithms within GAMS
- Interaction with other systems:
  - Job control
  - Data exchange

Model

- Platform
- Solver
- Data
- Interface
Independence of Model and Operating System

Platforms supported by GAMS:

⇒ Models can be moved between platforms with ease!
Independence of Model and Solver

One environment for a wide range of model types and solvers

- All major commercial LP/MIP solver
- Open Source Solver (COIN)
- Also solver for NLP, MINLP, global, and stochastic optimization

Switching between solvers with one line of code!
Independence of **Model and Data**

- Declarative Modeling
- ASCII: Initial model development

- GDX: Data layer ("contract") between GAMS and applications
  - Platform independent
  - No license required
  - Direct GDX interfaces and general API
  - ...

---

**Model**
- Platform
- Solver
- Data
- Interface
Independence of Model and User Interface

API’s

- Low Level
- Object Oriented: .Net, Java, Python
- No modeling capability: Model is written in GAMS
- Wrapper class that encapsulates a GAMS model
Smart Links to other Applications

- User keeps working in his productive tool environment
- Application accesses all optimization capabilities of GAMS through API
- Visualization and analysis of model data and results in the application
Smart Links to other Applications

- User keeps working in his productive tool environment
- Application accesses all optimization capabilities of GAMS through API
- Visualization and analysis of model data and results in the application
Smart Links to other Applications

- User keeps working in his productive tool environment
- Application accesses all optimization capabilities of GAMS through API
- Visualization and analysis of model data and results in the application
Models must benefit from:

- Advancing hardware / New Platforms
- Enhanced / new solver and solution technology
- Improved / upcoming interfaces to other systems
- New Modeling Concepts

Protect investments of Users

- Life time of a model: 15+ years
- New maintainer, platform, solver, user interface
- Backward Compatibility
- Software Quality Assurance

Striving for Innovation and Compatibility
Agenda

What is GAMS?

Enhancements in GAMS
- Singleton Sets
- Value at Risk
- GUSS & Grid
- Obfuscated Save/Restart Files

Enhancements around GAMS

Summary
**Singleton Sets**

- Special GAMS `Set`
- Has at most one element (empty **Singleton Sets** are also valid)
- No need to be controlled by controlling index nor indexed operator

```plaintext
Set      s   / s1*s3 /;
Set      multiS(s) / s2   /;
Parameter p(s);
Scalar   x;

p(s) = ord(s);
x = sum(multiS,p(multiS));

Set      s   / s1*s3 /;
Set      single(s) / s2   /;
Parameter p(s);
Scalar   x;

p(s) = ord(s);
x = p(single);
```
Singleton Sets

- Special GAMS Set
- Has at most one element (empty Singleton Sets are also valid)
- No need to be controlled by controlling index nor indexed operator
- Any assignment to a Singleton Set first clears or empties the Set, no explicit clear is necessary

```gams
Set     i   Static Set / a, b, c /
ii(i)   Dynamic Set / b /
Singleton Set si(i) Dynamic Singleton Set / b /
ii('c') = yes;
si('c') = yes;
Display ii, si;

----  8 SET ii  Dynamic Set
     b,  c

----  8 SET si  Dynamic Singleton Set
     c
```

→ Convenience
→ “Security”
Stochastic Programming

• The Extended Mathematical Programming (EMP) framework is used to replace parameters in the model by random variables

• Support for Multi-stage recourse problems and chance constraint models

• Easy to add uncertainty to existing deterministic models, to either use specialized algorithms or create Deterministic Equivalent (new free solver DE)

Simple **Newsboy** Problem

* LostSales = demand - UnitsSold
lSales.. L =e= d - S;
* Inventory = UnitsBought - UnitsSold
Inv.. I =e= X - S;
* Profit, to be maximized
Profit.. Z =e= r*S - c*X - h*I - p*L;

Model nb / all /;
Solve nb max z use lp;
Simple Newsboy Problem

* LostSales = demand - UnitsSold
1Sales.. L =e= d - S;
* Inventory = UnitsBought - UnitsSold
Inv.. I =e= X - S;
* Profit, to be maximized
Profit.. Z =e= r*S - c*X - h*I - p*L;

Set scen / s1*s6 /;
Parameter
    s_d(scen) Demand
    s_x(scen) Units bought;
Set
    dict / scen.scenario.''
          d .randvar .s_d
          x .level .s_x /;

Model nb / all /;
Solve nb max z use emp scenario dict;
Keywords

- Risk Measures
  - cVaR
    - Conditional Value at Risk
  - ExpectedValue
    - Expected Value
  - VaR
    - Value at Risk
- Chance
  - Chance Constraints
- JRandVar
  - Random Variables with joint distribution
- RandVar
  - Discrete and parametric random variables
- Sample
  - Customize samples taken from random variables with continuous distribution
- Stage
  - Stage of (random) variables and equations
GUSS & Grid

- Update model data instead of matrix coefficients/rhs
- Hot start (keep the model hot inside the solver and use solver’s best update mechanism)
- Save model generation and solver setup time
- Model rim unchanged from scenario to scenario
- Apriori knowledge of all scenario data
GUSS & Grid

- Scalable:
  - Support of massive grids, but also
  - Multi-CPU / Multiple cores desktop machines
- Platform independent
- Only minor changes to model required
- Separation of model and solution method
  → Model stays maintainable
Set cs(s) scenarios per GUSS run
    dict / cs.scenario.''
    d .param .dd
    f .param .ff
    x .level .xx /

Parameter h(cpu) grid handles;
transport.solverlink=%solvelink.AsyncGrid%;
loop(cpu,
    cs(s) = CpuSMap(cpu,s);
    solve mymodel min z using lp scenario dict;
    h(cpu) = subproblem.handle );

repeat
    loop(cpu$handlecollect(h(cpu)),
        display$handledelete(h(cpu)) 'trouble deleting handles';
        h(cpu) = 0
    );
    display$sleep(card(h)*0.02) 'sleep for some time';
until card(h)=0;
Obfuscated Save/Restart File

- Special Save/Restart File
- Symbol and UEL names are obfuscate
- New options `saveobfuscate (so)` and `xsaveobfuscate (xso)` to generate obfuscated Save/Restart file (regular or compressed)
- New option `restartNamed (rn)` to bring back original names when restarting from an obfuscated Save/Restart file
Obfuscated Save/Restart File

- Intended use:
  - Compile (only) GAMS model into named and obfuscated save file:
    $$\texttt{call gams trnsport a=c s=0 named saveobfuscate=0anon}$$
  - Move obfuscated save file to non-secure machine and execute it there:
    $$\texttt{call gams dummy r=0anon s=1anon gdx=demo}$$
Obfuscated Save/Restart File

- Intended use:
  - Compile (only) GAMS model into named and obfuscated save file:
    $call gams transport a=c s=0named saveobfuscate=0anon
  - Move obfuscated save file to non-secure machine and execute it there:
    $call gams dummy r=0anon s=1anon gdx=demo
  - Bring new (still obfuscated) save file with results back to safe machine and do continued compilation with reporting and export:
    $call gams dummy r=1anon restartNamed=0named gdx=res

➡️ Data Security
GENERAL ALGEBRAIC MODELING SYSTEM

Agenda

What is GAMS?

Enhancements in GAMS

Enhancements around GAMS
  - MINLPLib 2
  - GAMS Lessons

Summary
MINLPLib and GLOBALLib

- [http://www.gamsworld.org](http://www.gamsworld.org)
- Initiated in 2001 (as part of GamsWorld/MinlpWorld/GlobalWorld):
  - M. Bussieck, A. Drud, and A. Meeraus
  - MINLPLib – A Collection of Test Models for Mixed-Integer Nonlinear Programming
- Frequently used for testing, but also benchmarking
- Scalar GAMS format
- Varying from small scale (great for debugging!) to large scale real world instances (agricultural economics, chemical-, civil-, and electrical engineering, finance, management, OR)
- Intentionally including instances from badly formulated models or different formulations of the same problem
- Including solution points for many instances
MINLPLib 2

• Tasks:
  • Adding new problem instances:
    • Both convex and nonconvex problems
    • (MI)QPs, (MI)QCQPs, and (MI)NLPs
    • Easy solvable, solvable, difficult to solve, but not trivial
  • Categorizing instances
    • Convexity
    • Problem type (quadratic, polynomial, general nonlinear)
    • Function types (powers, exp/log, trigonometric, ...)
    • Solved to global optimality?
  • Providing feasible best known solutions

• Work in progress, alpha version publicly available:
MINLPLib 2 – New Instances

- Mainly from:
  - CMU-IBM open source MINLP project (convex MINLPs)
  - minlp.org
  - POLIP (polynomial MINLPs)

- Future Work:
  - Add more NLPs (merge in PrincetonLib, ...)
  - Semi-automatic identification of duplicates
  - More structure recognition, e.g., second-order cones
  - Define interesting subsets, especially a benchmark set for global solvers

If you have interesting instances, please consider contributing.
YouTube Channel: GAMS lessons

A Brief Introduction to Modeling in GAMS
4 months ago • 1,507 views
In this video we will review a simple transport model. We will also have a look at the output files, i.e. log and listing file, that is obtained after solving an instance of a model.

GAMS License File Installation and Component Review
4 months ago • 330 views
The GAMS license file is installed on GAMS 24.2.1. The license file content is reviewed by running GAMS model library example licmemo (#307).

How to Install the Native GAMS Version on a Mac
4 months ago • 329 views
We will make a apple script that launches a terminal window, which is set up in a way that GAMS keywords can be immediately used.

Install the Windows Version of GAMS on a Mac by Using Wine
4 months ago • 291 views
We will setup the graphical user interface for GAMS, GAMSIDE, on a Mac by using a third
Agenda

What is GAMS?

Enhancements in GAMS

Enhancements around GAMS

Summary
Summary

• Improvements on all frontiers:
  • Extended Syntax
    • Singleton Sets
  • Solution concepts
    • Stochastic Programming
  • Multithreading
    • Guss/Grid
  • Data Security
    • Obfuscated Save/Restart files
  • Quality Assurance and Benchmarking
    • MINLPLib 2
  • Documentation
    • YouTube Channel
Fields of Fuel - A Multiplayer, Web-based Simulation Game

A complex system of GAMS models is a centerpiece of this free web-based simulation game, which allows players to explore sustainability issues associated with bioenergy crop production. Biofuels and agronomic experts assisted in creating an accurate and realistic depiction of the system dynamics.

- Players take on the role of farmers working to sustainably grow crops as energy resources, earn income and improve ecosystem services.
- Automated ‘bot’ players communicate with the optimization models via the GAMS Java API to evaluate which options will maximize their overall game score.
- The game can be played in a variety of settings, but was primarily designed for use in high school and undergraduate classes.

For further information please visit http://www.fieldsoffuel.org
or contact Steven Wangen - srwangen@wisc.edu