Software Tutorial

Pete Steacy  PSteacy@gams.com
GAMS Development Corp.
www.gams.com

Huntington Beach, CA  April 2012
Agenda

- What is GAMS?
- What is new?
Agenda

What is GAMS?

- GAMS at a Glance
- A simple Example
- Interfacing with other Applications
What does this modeler have to think about?

- Application
- Mathematics
- Computer Code

GAMS eases the transitions between these domains.
GAMS Development / Software at a Glance

- Roots: World Bank, 1976
- Went commercial in 1987
- GAMS Development Corp. (US)
- GAMS Software GmbH (Europe)
- Technical tool provider (Software)
- Broad academic & commercial user community and network
  - GAMS is used in more than 120 countries
  - Half of licenses commercially used
Broad Academic and Commercial Network

ClustrMaps archive for http://www.gams.com/download/

5177 visits from 19 Mar 2012 to 26 Mar 2012

Distance in which individuals are clustered
Total number of visits depicted above = 4275

Dot sizes:
- \( \bullet \) = 1000 +
- \( \bullet \) = 100 - 999
- \( \bullet \) = 10 - 99
- \( \bullet \) = 1 - 9
Downloads (March 2012)

<table>
<thead>
<tr>
<th>Region</th>
<th>GB Transfer Out</th>
<th>GB Requests</th>
<th>Requests</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>United States</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>First 10 TB</td>
<td>197,126</td>
<td>3</td>
<td>23.66</td>
</tr>
<tr>
<td>HTTPS Requests</td>
<td>52,154</td>
<td>0.01</td>
<td>0.04</td>
</tr>
<tr>
<td>HTTP Requests</td>
<td></td>
<td></td>
<td>23.71</td>
</tr>
<tr>
<td><strong>Europe</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>First 10 TB</td>
<td>212,982</td>
<td>1</td>
<td>25.56</td>
</tr>
<tr>
<td>HTTPS Requests</td>
<td>16,456</td>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>HTTP Requests</td>
<td></td>
<td></td>
<td>25.58</td>
</tr>
<tr>
<td><strong>Asia Pacific (Tokyo)</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>First 10 TB</td>
<td>23,800</td>
<td>4.78</td>
<td></td>
</tr>
<tr>
<td>HTTPS Requests</td>
<td>4,676</td>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>HTTP Requests</td>
<td></td>
<td></td>
<td>4.79</td>
</tr>
<tr>
<td><strong>Asia Pacific (Singapore)</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>First 10 TB</td>
<td>39,512</td>
<td>7.51</td>
<td></td>
</tr>
<tr>
<td>HTTPS Requests</td>
<td>18,087</td>
<td>0.02</td>
<td>0.02</td>
</tr>
<tr>
<td>HTTP Requests</td>
<td></td>
<td></td>
<td>7.54</td>
</tr>
<tr>
<td><strong>South America</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>First 10 TB</td>
<td>21,656</td>
<td>5.41</td>
<td></td>
</tr>
<tr>
<td>HTTPS Requests</td>
<td>1,535</td>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>HTTP Requests</td>
<td></td>
<td></td>
<td>5.42</td>
</tr>
</tbody>
</table>

Total: 495 GB ~ 5,500 monthly downloads
Agenda

- What is GAMS?
- GAMS at a Glance
- A simple Example
- Interfacing with other Applications
GAMS at a Glance

- Balanced mix of declarative and procedural elements
- Platform independence
- Hassle-free switch of solution methods
- Open architecture and interfaces to other systems
- Independent layers
GAMS at a Glance: Balanced mix...

**Balanced mix of declarative and procedural elements**
- Algebra (Expressions): model equations
- Relational Algebra (SQL) for data manipulation

**Mathematical Notation**

\[
\sum_{c,p: (c,p) \in N} t_{cost} \cdot dist(c, p) \cdot x_c^p \rightarrow \min
\]

\[
\sum_{c,p: (c,p) \in N} x_c^p \leq sup(c) \quad \forall c
\]

\[
\sum_{c,p: (c,p) \in N} x_c^p \geq dem(p) \quad \forall p
\]

\[
x_c^p \geq 0 \quad \forall c, p : (c, p) \in N
\]

**Notation in GAMS**

```
Equations
cost       define objective function
supply(i)  observe supply limit at plant i
demand(j)   satisfy demand at market j;

cost       ..    z =e= sum((i,j), c(i,j)*x(i,j)) ;
supply(i)   ..    sum(j, x(i,j)) =l= a(i) ;
demand(j)   ..    sum(i, x(i,j)) =g= b(j) ;

Model transport /all/ :
Solve transport using lp minimizing z ;
Display x.l, x.m ;
```
GAMS at a Glance: Balanced Mix..

Balanced mix of declarative and procedural elements

- For

  ```
  scalar scen;
  for (scen=1 to 10 by 0.5,
    f   = 10*scen;
    c(i,j) = f * d(i,j) / 1000;
  Solve transport using lp minimizing Z;
  Display Z.l;);
  ```

- Loop/If

  ```
  loop(h,
    if (work(h),
      pay(i,h) = 0.6*pay(i,h);
    else
      pay(i,h) = 1.5*pay(i,h);
    )
  );
  ```

- While

  ```
  scalar scen /1/;
  while (scen<=10,
    f   = 10*scen;
    c(i,j) = f * d(i,j) / 1000;
  Solve transport using lp minimizing Z;
  scen = scen + 0.5;
  );
  ```

- Macros

- User defined function libraries
GAMS at a Glance

- Balanced mix of declarative and procedural elements
- **Platform independence**
- Hassle-free switch of solution methods
- Open architecture and interfaces to other systems
- Independent Layers

10+ Supported Platforms

- Solaris 64
- AIX
- Linux 32
- Linux 64
- Mac
- Windows 32
- Windows 64
- HP
GAMS at a Glance

- Balanced mix of declarative and procedural elements
- Platform independence
- Hassle-free switch of solution methods
- Open architecture and interfaces to other systems
- Independent Layers
GAMS at a Glance

• Balanced mix of declarative and procedural elements
• Platform independence
• Hassle-free switch of solution methods
• Open architecture and interfaces to other systems
• Independent Layers

• ASCII
• Gams Data eXchange (Binary)
  • MS Excel, Access
  • Databases
  • ....
• API’s
• Component Libraries
• .NET Integration
GAMS at a Glance

**Independence of**
- Model and data
- Model and solution methods (solver)
- Model and operating system
- Model and user interface

→ **Models benefit from**
- Advancing hardware
- Enhanced / new solver technology
- Improved / upcoming interfaces to other systems
GAMS at a Glance: Development Environment

GAMS IDE
• Project management
• Editor / Syntax coloring / Spell checking
• Launching and monitoring of (multiple) GAMS processes
• Listing file / Tree view / Syntax-error navigation
• Solver selection / Option selection
• GDX viewer
  – Data cube
  – Data export (e.g. to MS Excel)
  – Charting facilities
• Model libraries
• Documentation
# Agenda

<table>
<thead>
<tr>
<th>What is new?</th>
</tr>
</thead>
<tbody>
<tr>
<td>GAMS System</td>
</tr>
<tr>
<td>Platforms</td>
</tr>
<tr>
<td>Solvers</td>
</tr>
<tr>
<td>Interfaces</td>
</tr>
<tr>
<td>Stochastic Programming</td>
</tr>
</tbody>
</table>
What is new: GAMS System

• Support for user-defined
  – Macros
  – Function libraries
  – External equations

• Asynchronous execution

• Extended Mathematical Programming (EMP)

• More and further details: http://www.gams.com/docs/release/release.htm
What is new: Asynchronous Execution

- \$Call / Execute / put_utility 'Exec':
  - Start jobs from GAMS

- Three ways to start a job asynchronously,
  - at compile time (CT):
    - \$Call.ASync ...
  - at execution time (ET):
    - Execute.Async '...';
    - put_utility 'Exec.ASync' / '...';
What is new: Platforms

• Support for MAC OS X
• Cross-platform licenses
• Wine (Linux, Mac)
What is new: Solvers

- GloMIQO: Branch-and-bound global optimization for mixed-integer quadratic models
- Lindo: Global and stochastic optimization
- Gather-Update-Solve-Scatter
- (Stochastic) EMP
What is new: Gather-Update-Solve-Scatter

<table>
<thead>
<tr>
<th>Setting</th>
<th>Solve time (secs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solvelink=0 (default)</td>
<td>40.297</td>
</tr>
<tr>
<td>Solvelink=%Solvelink.LoadLibrary%</td>
<td>03.625</td>
</tr>
<tr>
<td>GUSS</td>
<td>00.797</td>
</tr>
</tbody>
</table>

- Updates model data instead of matrix coefficients/rhs
- Hot start (keeps the model hot inside the solver and uses solver’s best update mechanism)
- Saves model generation and solver setup time
- Model unchanged from scenario to scenario
- A priori knowledge of all scenario data
What is new: Solving Data Related Models

The common way:

Set s / s1*s10 / 
Parameter 
   A_s(s,i,j) Scenario data 
   xlo_s(s,j) Scenario lower bound for variable x 
   xl_s(s,j) Scenario solution for x.l 
   em_s(s,i) Scenario solution for e.m;

Loop(s, 
   A(i,j) = A_s(s,i,j); 
   x.lo(j)= xlo_s(s,j); 
   solve mymodel min z using lp; 
   xl_s(s,j) = x.l(j); 
   em_s(s,i) = e.m(i); 
);

• GAMS generates model and writes it to hd
• GAMS writes database to scratch files on hd
• GAMS calls solver and vacates memory
• After solver is done: GAMS restarts and swaps database
The new way:

Set s / s1*s10 /
Parameter
  A_s(s,i,j) Scenario data
  xlo_s(s,j) Scenario lower bound for variable x
  xl_s(s,j) Scenario solution for x.l
  em_s(s,i) Scenario solution for e.m;
Set dict / s. scenario. ' '
  A. param. A_s
  x. lower. xlo_s
  x. level. xl_s
  e. marginal. em_s /
solve mymodel min z using lp scenario dict;
What is new: Interfaces

- API’s for various programming languages (C, Fortran, Delphi)
- Component libraries
- Better integration into Python
What is new: Distributed GAMS APIs

- Component Libraries
  - GAMS
  - GDX
  - Option

- Supported languages
  - C, C++, C#
  - Delphi
  - Fortran
  - Java
  - VBA, VB.Net
  - Python

- Examples/Documentation
What is new: GAMS Component Libraries
What is new: Calling GAMS from Python

```python
if __name__ == "__main__":
    numberParams = len(sys.argv)
    if numberParams != 2:
        print("Usage:", sys.argv[0], "sysDir"
        os._exit(1)

gdxHandle = new_gdxHandle_tp()
optHandle = new_optHandle_tp()
gamsxHandle = new_gamsxHandle_tp()

sysDir = sys.argv[1]
print(sys.argv[0], "using GAMS system directory:", sys.argv[1]

assert gamsxCreateD(gamsxHandle, sysDir, GMS_SSSIZE)[0]
assert gdxCreateD (gdxHandle, sysDir, GMS_SSSIZE)[0]
assert optCreateD (optHandle, sysDir, GMS_SSSIZE)[0]

status = writeModelData(gdxHandle, "demanddata.gdx")
if not status:
    print("Model data not written")
    terminate(gdxHandle, gamsxHandle, optHandle)

status = callGams(gamsxHandle, optHandle, sysDir)
if not status:
    print("Call to GAMS failed")
    terminate(gdxHandle, gamsxHandle, optHandle)

status = readSolutionData(gdxHandle, "results.gdx")
if not status:
    print("Could not read solution back")
    terminate(gdxHandle, gamsxHandle, optHandle)
```

Creating Input for GAMS Model

Callout to GAMS

Reading Solution from GAMS Model
What is new: .Net Integration

```csharp
using System;
using System.Collections.Generic;
using System.Text;
using GAMS;

namespace TransportSeq
{
    class Transport1
    {
        static void Main(string[] args)
        {
            GAMSWorkspace ws = new GAMSWorkspace();
            ws.GamsLib("trnsport");
            using (GAMSJob t1 = ws.AddJobFromFile("trnsport.gms"))
            {
                t1.Run();
                Console.WriteLine("Ran with Default");
                foreach (GAMSVariableRecord rec in t1.OutDB.GetVariable("x"))
                {
                }
                using (GAMSOptions opt = ws.AddOptions())
                {
                    opt.AllModelTypes = "xpress";
                    t1.Run(opt);
                }
                Console.WriteLine("Run with XPRESS");
                foreach (GAMSVariableRecord rec in t1.OutDB.GetVariable("x"))
                {
                }
            }
        }
    }
}
```
## Agenda

<table>
<thead>
<tr>
<th>What is new?</th>
</tr>
</thead>
<tbody>
<tr>
<td>GAMS System</td>
</tr>
<tr>
<td>Platforms</td>
</tr>
<tr>
<td>Solvers</td>
</tr>
<tr>
<td>Interfaces</td>
</tr>
<tr>
<td>Stochastic Programming</td>
</tr>
</tbody>
</table>
Stochastic Programming Claims and ‘Facts’

- Lots of application areas (Finance, Energy, Telecommunication)
- Mature field (Dantzig ’55)
- Variety of SP problem classes with specialized solution algorithms (e.g. Bender’s Decomposition)

- Small fraction compared to deterministic mathematical programming (NEOS)
- Few commercially supported solvers for SP
- Various frustrations with industrial SP projects
Extended Mathematical Programming

- Embedded Complementarity Systems
- Disjunctive Programs
- Bilevel Programs
- Extended Nonlinear Programs
- Stochastic Programming
- ...

- Breakouts of traditional MP classes
- No conventional syntax
- Limited support with common model representation
- Incomplete/experimental solution approaches
- Lack of reliable/any software
Thank You!

USA
GAMS Development Corp.
1217 Potomac Street, NW
Washington, DC 20007
USA
Phone: +1 202 342 0180
Fax: +1 202 342 0181
http://www.gams.com
sales@gams.com
support@gams.com

Europe
GAMS Software GmbH
Eupener Str. 135-137
50933 Cologne
Germany
Phone: +49 221 949 9170
Fax: +49 221 949 9171
http://www.gams.com
info@gams.de
support@gams.com