Rapid Application Prototyping with GAMS

Steven Dirkse
sDirkse@gams.com
GAMS Development Corp
www.gams.com

Alex Meeraus
AMeeraus@gams.com
GAMS Development Corp
www.gams.com

INFORMS International
Puerto Rico, 10 July, 2007
Welcome/Agenda

- GAMS Development / GAMS Software
- Working with GAMS – A Guided Tour
- Model Development
- Model Deployment and Maintenance
<table>
<thead>
<tr>
<th>Agenda</th>
</tr>
</thead>
<tbody>
<tr>
<td>GAMS Development / GAMS Software</td>
</tr>
<tr>
<td>Working with GAMS – A Guided Tour</td>
</tr>
<tr>
<td>Model Development</td>
</tr>
<tr>
<td>Model Deployment and Maintenance</td>
</tr>
</tbody>
</table>
GAMS Development / GAMS Software

• Roots: Research project
World Bank 1976
• Pioneer in Algebraic
Modeling Systems
used for economic modeling
• Went commercial in 1987
• Offices in Washington, D.C
and Cologne

• Professional software tool
provider
• Operating in a segmented
niche market
• Broad academic &
commercial user base
and network
## 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>

* Illustrative examples in the GAMS Model Library
Agenda

- GAMS Development / GAMS Software
- Working with GAMS – A Guided Tour
- Model Development
- Model Deployment and Maintenance
GAMS at a Glance

General Algebraic Modeling System:

Design Principles:
• Balanced mix of declarative and procedural elements
• Open architecture and interfaces to other systems
• Different layers with separation of:
  – model and data
  – model and solution methods
  – model and operating system
  – model and interface
System Overview

Connectivity Tools
- Uniform Data Exchange:
  - ASCII
  - GDX (ODBC, SQL, XLS, XML)
- GDX Tools
- Data API
- Ext. programs
  - EXCEL
  - MATLAB
  - GNUPLOT, ...
  - C, Delphi, ...

GAMS Language Compiler and Execution System

User Interfaces

Solvers
LP-MIP-QCP-MIQCP-NLP-MINLP-CNS-MCP-MPEC
MPSGE, global, and stochastic optimization

Interactive API/Batch

Productivity Tools
- Integrated Development Environment (IDE)
- Model Debugger and Profiler
- Model Libraries
- Data Browser
- Charting Engine
- Benchmarking
- Deployment System
- Quality Assurance and Testing

BARON, COIN, CONOPT, CPLEX, DECIS, DICOPT, KNITRO, LGO, MINOS, MOSEK, OQNLP, PATH, SNOPT, XA, XPRESS, ...
Hands-on! Installing GAMS

Welcome to the GAMS 22.2 Setup Wizard

Select Components
Which components should be installed?

Full installation: yes
McCalm User Guide: no

Copy license file

lewis.gams.com - PuTTY

archive: euro06lnx.zip
inflating: 1x3gams_sfx.exe

UnZipSFX 5.41 of 16 April 2000, by Info-ZIP (Zip-Bugs@lists.wku.edu).

extracting: gams.zip
inflating: gamsinst
inflating: gamsunpak
inflating: gamsunzip

$rm euro06lnx.zip 1x3gams_sfx.exe
$gamsinst -a

gamsinst version 034
----------------------
Installation of GAMS distribution 22.2

Unpecking GAMS ... estimated disk blocks needed: 20480, available: 923181
executing--> ./gamsunpak

$export PATH=/home/susanne/euro2006/:$PATH
Hands-on! Testing the installation

This file contains the basic data and definition of the surface water system. Data is complete for year 1988. Some parameters could be computed for future years using growth rates provided in this file, others had to be estimated and entered. Enter the year for which the setup is desired in set 1sr (set 1sr should have only one entry).


Changes for year 2000 runs:
- Growth of crop yields set to a maximum of 3%
- Insert this line after growthc parameter:

```
  *growthc = growthc + growthc rate * yield
```

Objective:

```
  INDUS89.gms
```

--- Restarting execution
--- indus89.gms(3621) 0 Mb
--- Reading solution for model wsbn
--- indus89.gms(3621) 4 Mb
--- Status: Normal completion
--- Job indus89.gms Stop 06/29/06 04:58:12 elapsed: 0:00:02.774
Hands-on! Testing the installation

GAMS

lewis.gams.com - PuTTY

$gamslib indu89
Model indu89.gms retrieved
$gams indu89
--- Job indu89 Start 06/29/06 05:01:20
GAMS Rev 145 Copyright (C) 1987-2006 GAMS Development. All rights reserved
Licensee: EURO 2006 GAMS Workshop 0060626/00C1CB-LNX
GAMS Software GmbH DC5946
--- Starting compilation
--- indu89.gms(3622) 4 Mb
--- Starting execution
--- indu89.gms(3618) 5 Mb
--- Generating LP model wizsa
--- indu89.gms(3621) 7 Mb
--- 2,726 rows 5,570 columns 39,489 non-zeroes
--- Executing CPLEX

GAMS/Cplex Apr 21, 2006 LNX.CF.CF 22.2 031.034.041.LX3 For Cplex 10.0
Cplex 10.0.1, GAMS Link 31
Cplex licensed for 1 use of Ip, qp, mip and barrier, with 4 parallel threads.

Reading data...
Starting Cplex...
Tried aggregator 1 time.
LP Presolve eliminated 230 rows and 805 columns.
Aggregator did 552 substitutions.
Reduced LP has 1794 rows, 5113 columns, and 33006 nonzeros.
Presolve time = 0.04 sec.
Initializing dual steep norms . . .

Iteration log . . .
Iteration: 1 Scaled dual infeas = 29555.67467575
A few Words about GAMS Syntax

Minimize Transportation cost
subject to Demand satisfaction at markets
       Supply constraints
\[
\sum_{c,p: (c,p) \in N} tcost \cdot dist(c,p) \cdot x^c_p \rightarrow \min \\
\sum_{c,p: (c,p) \in N} x^c_p \leq sup(c) \quad \forall c \\
\sum_{c,p: (c,p) \in N} x^c_p \geq dem(p) \quad \forall p \\
x^c_p \geq 0 \quad \forall c, p : (c, p) \in N
\]
GAMS Syntax – GAMS Algebra

Variables
  x(i,j)  shipment quantities in cases
  z       total transportation costs in thousands of dollars;

Positive Variable x ;

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/ ;
GAMS Syntax – cont.

• Symbols:
  - Sets
  - Parameters
  - Variables
  - Equations
  - Models
  - ASCII Output Files

Set I some stuff /cat,dog,ding1*ding10/
Parameter life(I) life count / cat 7 /
Integer Variable x(I) number to purchase;
Equation e(I) relate something;
Model animallife /e, some, more/;
File fx some file /'c:\t\text.txt' /

• Statements
  - Declaration+Data statement
  - Data Assignments
  - Equation Definition
  - Programming Flow Control
  - Option statement

Set I /cat,dog/;
life('dog')=life('cat')-1; x.lo(I)=1;
e(I).. Sqr(x(I)) =l= log(life(I));
loop(I, put fx I.tl);
Option reslim=10;
Hands-On! Inspect trnsport.gms

• IDE:
  File → Model Library
  trnsport
  Hit F9 or Click

• Unix:
  $ gamslib trnsport
  $ vi trnsport.gms
  $ gams trnsport
  $ vi trnsport.lst
Hands-on! IDE - A Guided Tour

- IDE Project Management
- Documentation
- Model Library
- Editor
- Solver Selection
- Option Selection
- Listing file/Tree view/Error navigation
- GDX Viewer
  - Data cube
  - Export to Excel
  - Graphs
Welcome to the Daily SuDoku!

Today’s SuDoku is shown on the right. Click the grid to download a printable version of the puzzle. Visit the archive for previous daily puzzles and solutions. Play online, print a SuDoku, solve and get hints using the new improved Draw/Play function.

But how do I do it?

The object is to insert the numbers in the boxes to satisfy only one condition: each row, column and 3x3 box must contain the digits 1 through 9 exactly once. What could be simpler?

The rules of the new Monster SuDoku are exactly the same, but more numbers and letters are needed.
Christmas tree Sudoku

Daily SuDoku

Daily Seasonal Sudoku: Fri 23-Dec-2005  [instructions]
Hands-on! Basic Sudoku (su1 -> su2)

- Basic model su1 computes solution to Sudoku
- Is the solution unique?
- If not, how many solutions exist?
- Edits for su1 -> su2:
  - Implement binary cuts to exclude known solutions
  - To cut off xb: sum(i, abs(xb(i)-x(i)) =g= 1;
  - Use loop to find and store solutions
  - Use GDX to view model data and solution
Input/Output through ASCII Files

• ASCII Input Data
  – Part of model input ($\texttt{include file.txt}$)
  – Posix Utilities are part of GAMS Windows System
    • Platform independent data file preparation
    • $\texttt{sed, awk, grep, cut, ...}$
      $\texttt{$\textbackslash{call\ cut\ -d,\ -f1,3-\ file.txt\ >\ filenew.txt}$}$

• ASCII File Output
  – GAMS Put Facilities
GAMS Data eXchange

- **GAMS Data eXchange (GDX):**

  ![Diagram of data exchange process]

  - Complements the ASCII text data input
  - Advantages:
    - Fast exchange of data (factor >20)
    - Syntactical check on data before model starts
    - Compile-time and Run-time Data Exchange
    - Platform Independent
GDX Tools

IDE
GDX Viewer

gdxsplit

gdxmerge

gdxdump

gdxdiff

GDX

GDX API

gams

gdxxrw
(MS Office)
Hands-on! Report on all solutions (su3)

- Find a “good” cell to fix
  - A good fix is one that leaves few (one?) solutions
  - Compute result of any fix using saved GDX data
Hands-on! Force Uniqueness (su2 -> su4)

• Set cell r7.c4 to the value 4 (c.f. su3)
• Verify that the resulting solution is unique
### Hands-on! Text output (rep0 - rep3)

<table>
<thead>
<tr>
<th></th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>7</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>7</td>
<td>2</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
</tbody>
</table>

Hands-on! Infeasible Sudoku (su1 -> su5)

- What should we do with an infeasible Sudoku?
  - Not enough to just report the infeasibility
  - Here, repair the data to make the model feasible
- Edits for going from su1 -> su5
  - Add binary variable undo (relaxes fixed cells)
  - Remove x.fx for fixed cells
  - Use random generation to get bogus data
Hands-on! Mapping data (map1)

• We solve the Samurai as 5 basic puzzles, with linking constraints for the overlapping cells
• Requires mapping 21x21 Samurai puzzle into separate 9x9 puzzles
Hands-on! Samurai model (su5 -> su6)

- Add puzzle index p to all variables/equations
- Add linking constraints
- Use random data to test
- Fix undo variables initially to 0
  - If the model is feasible, it will solve quickly
  - If infeasible, we unfix undo and resolve
Agenda

- GAMS Development / GAMS Software
- Working with GAMS – A Guided Tour
- Model Development
- Model Deployment and Maintenance
Important Principles

- Deployed models have often 15+ years lifecycle
  - Changing environment:
    - hardware, operating system, interface (GUI/data)

- Backward compatibility
- Platform/Solver/Interface Independence
  - Model benefits from
    - Advanced hardware
    - Advanced solver technology

- Reduced Total Cost of Ownership (TCO)
Flow of Data

Data Model I
- Application in control of data processing
- No direct data access

Data Model II
- Large Scale/Raw data exchange GAMS↔DB
- Control Data GAMS↔Application
Hands-on! Excel in charge (samurai_vb)

- Existing Samurai model with Excel GUI
- Look at data communication between model and GUI
Hands-on! Samurai data input (su6 -> su7)

• Prepare our Samurai model su6 to plug in to spreadsheet
• Import/export 21x21 data from GUI (via GDX)
• Use mappings from map1 to map 21x21 -> 5x9x9
• Validate solution produced via GDXDIFF
  – Compare to solution from old application
Hands-on! Clean up (su7 -> su8)

- Create text file for display in GUI
A few Words about Maintenance

**Optimization**
- Takes Longer than one is willing to wait
- It will eventually fail

**Application**
- Real Time
- Always need a Solution to Problem

- Key for support/maintenance
  - Catch problems before a model is solved
    - Implement Data Error checks
  - Reproduce the problem offline
    - Get hold of Instance (`dumpopt=11`)
  - Solver related problems in confidential models
    - Get scalar Model using solver `CONVERT`
Summary

- 30+ Years Experience in Modeling
  - Strong views on modeling process (*The GAMS Way*)
    - Development
    - Deployment
    - Maintenance
  - Less than 5% of modeling/optimization projects do not fit the GAMS way
  - Use of GAMS and its productivity tools (after potentially steep learning curve)
    - Increases productivity of model building
    - Reduces total cost of ownership for model client
    - Opens doors to a large network of GAMS developers and clients with modeling needs
Contacting GAMS

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

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