Models and Their Roles

or

A Model is a Model is a Model*

Michael R. Bussieck
GAMS Development Corporation

Franz Nelissen
GAMS Software GmbH

GOR Workshop “Mathematical Optimization Services in Europe”
Bad Honnef, Germany October 14-15, 2004

* Freely adapted from the poetry of Gertrude Stein, 1874-1946, American writer
Agenda

• What is GAMS
• What is a GAMS Model
• Roles of a Model
  – Communication Vehicle
  – Analytic Framework
  – Cost Saver
• Conclusions
GAMS Overview

• Started as a Research Project at the World Bank 1976
• GAMS went commercial in 1987
• Opened European Office in Cologne, Germany 1996
• 10,000s of customers in over 100 countries
Basic Principles

• Separation of model and solution methods
• Model is a database operator and/or object
• Balanced mix of declarative and procedural approaches
• Computing platform independence
• Multiple model types, solvers, platforms
Multiple model types

- LP Linear Programming
- MIP Mixed Integer Programming
- NLP Nonlinear Programming
- MCP Mixed Complementarity Programming
- MINLP Mixed Integer Nonlinear Programming
- MPEC NLP with Complementarity Constraints
- MPSGE General Equilibrium Models
- Stochastic Optimization
Minimize: Transportation cost (distance & units)
Subject to: Demand satisfaction at markets
Supply constraints
GAMS Implementation

- Using the GAMS IDE to build a model
- Data Entry
- Max/Min Shipments
- Nonlinear Cost
- call GAMS IDE
Sets i / seattle, san-diego / 
    j / new-york, chicago, topeka / ;

Parameters a(i) / seattle 350, san-diego 600 /
    b(j) / new-york 325, chicago 300, topeka 275 /;

Table d(i,j) distance in thousands of miles

    new-york     chicago     topeka
  seattle       2.8         1.7        1.9
  san-diego     2.5         1.2        1.4 ;

Scalar f freight in dollars per case per thousand miles /90/ ;

Parameter rate(i,j); rate(i,j) = f * d(i,j) / 1000 ;
Model m1.gms

sets
i     canning plants
j     markets

parameters
a(i)     capacity of plant i in cases
b(j)     demand at market j in cases

Parameters:
c(i, j) transport cost in thousands of dollars per case

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) .. 
  a(i)  =g=  sum(j, x(i, j)) ;
demand(j) .. 
  sum(i, x(i, j)) =g=  b(j);

Model m1 /all/ ;
Model m1.gms (cont.)

```plaintext
model m1 /all/ ;

$call gams dat1 gdx=dat1 $gdxin dat1 $load i j a b c=rate

*--- solve LP and store results

solve m1 us lp min z ;

parameter rep(i,j,*) Summary Report;

rep(i,j,'lp') = x.l(i,j);
```
Min/Max Shipments

* min and max shipments

option limcol=0,limrow=0;

scalars xmin / 100 /
       xmax / 275 /;

binary variables ship(i,j)    decision variable to ship

equations    minship(i,j) minimum shipments
             maxship(i,j) maximum shipments ;

minship(i,j).. x(i,j) =g=   xmin*ship(i,j);
maxship(i,j).. x(i,j) =l=   xmax*ship(i,j);

model m2 min shipmenst / cost, supply, demand, minship, maxship /;
solve m2 using mip minimizing z;

rep(i,j,'mip') = x.l(i,j); display rep;
* nonlinear cost

equation nlcost nonlinear cost function; scalar beta;

nlcost.. z =e= sum((i,j), c(i,j)*x(i,j)**beta);
model m3 / nlcost,supply,demand /;

beta = 1.5; solve m3 using nlp minimizing z;
rep(i,j,'nlp-convex') = x.l(i,j);

beta = 0.6; solve m3 using nlp minimizing z;
rep(i,j,'nlp-non') = x.l(i,j);

option nlp=baron; solve m3 using nlp minimizing z;
rep(i,j,'nlp-baron') = x.l(i,j); display rep;
* min/max and nl obj

model m4 / nlcost,supply,demand, minship,maxship /;

option minlp=baron; solve m4 using minlp minimizing z;
option nlp=snopt; option optcr=0;
option minlp=sbb; solve m4 using minlp minimizing z;

rep(i,j,'minlp') = x.l(i,j); display rep;
What is a Model?

• List of Equations
  – Mathematical Programming (MP) Model

• Collection of several intertwined (MP) Models
  – Data Preparation and Calibration
  – “Solution” Module
  – Reporting Module

• Categorization of Models by answering:
  – Who is the User of a Model?
We are not Consultants

- No active acquisition of projects
- Extended User Support
- Projects with long time “friends”
- Help our clients out, if they are in “trouble”
Communication Vehicle

• Defining scope of a (part of a) project/model
• IT, analysts, managers, model builders have different views
• Misunderstandings common with verbal descriptions
• Use a model to define the scope
• Requirements for such a model
  – Rapid prototyping (max. 1-2 man days)
  – Standard IO interface (Excel)
  – Remote execution (Model Server)
Example

- Project in 2002 with large automotive company, scheduling of design verifications (tests)
- Replacement of I2 “off-the-shelf” scheduling tool, with customized model
- Scope defining model prototype
  - Built during first project meeting (<300 LOC)
  - Required data and output reports in spreadsheet
  - Model execution via email based GAMS remote application server (GRAS)
<table>
<thead>
<tr>
<th>A</th>
<th>H</th>
<th>I</th>
<th>J</th>
<th>K</th>
<th>L</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>tnetid</td>
<td>Site</td>
<td>*Requestor</td>
<td>*Category</td>
<td>*Procedure</td>
</tr>
<tr>
<td>318W906 ACCRO A1 NS06 0%</td>
<td>WD3PVQ</td>
<td>APTL</td>
<td>pkuchta</td>
<td>EMISSIONS</td>
<td>NS06</td>
</tr>
<tr>
<td>309W512-50K 2004 W/COMMON CWD3QE9</td>
<td></td>
<td>APTL</td>
<td>pkuchta</td>
<td>EMISSIONS</td>
<td>COMBO</td>
</tr>
<tr>
<td>ST01 EPA URBAN DRIVING SCHEIWD3QM5</td>
<td></td>
<td>APTL</td>
<td>pkuchta</td>
<td>DIESEL</td>
<td>EPA75_D</td>
</tr>
<tr>
<td>FWD-301W854 NS 67 HOT RESTAWD3P09</td>
<td></td>
<td>APTL</td>
<td>pkuchta</td>
<td>EMISSIONS</td>
<td>NS67</td>
</tr>
<tr>
<td>318W906 ACCRO A1 NS06 2%</td>
<td>WD3PVR</td>
<td>APTL</td>
<td>pkuchta</td>
<td>EMISSIONS</td>
<td>NS06</td>
</tr>
<tr>
<td>302T302-50K 2003 W/SIGMA FINWD3QED</td>
<td></td>
<td>APTL</td>
<td>pkuchta</td>
<td>EMISSIONS</td>
<td>COMBO</td>
</tr>
<tr>
<td>EPA HIGHWAY DRIVE CYCLE</td>
<td>WD3QM6</td>
<td>APTL</td>
<td>pkuchta</td>
<td>DIESEL</td>
<td>HWFET_D</td>
</tr>
<tr>
<td>FWD 560 ACCRO A US06 30TA4G WD3N64</td>
<td></td>
<td>APTL</td>
<td>ehunsang</td>
<td>EMISSIONS</td>
<td>NS103</td>
</tr>
<tr>
<td>FWD-560 ACCRO A NS06 2% 30T/WD3N65</td>
<td></td>
<td>APTL</td>
<td>ehunsang</td>
<td>EMISSIONS</td>
<td>NS06</td>
</tr>
<tr>
<td>ZE1H00 AUTO NS06 0% GRADE MWD3N66</td>
<td></td>
<td>APTL</td>
<td>ehunsang</td>
<td>EMISSIONS</td>
<td>NS06</td>
</tr>
<tr>
<td>FWD-NS61 CUST.VEH MZ125</td>
<td>WD3NKB</td>
<td>APTL</td>
<td>ehunsang</td>
<td>EMISSIONS</td>
<td>NS61</td>
</tr>
<tr>
<td>VEH. NO. 201888 - TWIN ROLLS</td>
<td>WD3NLG</td>
<td>APTL</td>
<td>ehunsang</td>
<td>EMISSIONS</td>
<td>COMBO</td>
</tr>
<tr>
<td>308W067 ACCRO C1 NS06 0%</td>
<td>WD3RAY</td>
<td>APTL</td>
<td>ehunsang</td>
<td>EMISSIONS</td>
<td>NS06</td>
</tr>
<tr>
<td>310W484 20F COLD CO</td>
<td>WD3HT5</td>
<td>APTL</td>
<td>ehunsang</td>
<td>EMISSIONS</td>
<td>NS77</td>
</tr>
<tr>
<td>FWD-EPA FUEL ECONOMY - HON WD3HU5</td>
<td></td>
<td>APTL</td>
<td>ehunsang</td>
<td>EMISSIONS</td>
<td>COMBO</td>
</tr>
<tr>
<td>ZE1H01 MANUAL NS06 0% GRADWD3N68</td>
<td></td>
<td>APTL</td>
<td>ehunsang</td>
<td>EMISSIONS</td>
<td>NS06</td>
</tr>
<tr>
<td>FWD-EPA75 W/MODAL CUST.VEHWD3NKD</td>
<td></td>
<td>APTL</td>
<td>ehunsang</td>
<td>EMISSIONS</td>
<td>75CVS</td>
</tr>
<tr>
<td>50K ACCRO#1 NS88 373/4X4 590TWD3NLK</td>
<td></td>
<td>APTL</td>
<td>ehunsang</td>
<td>EMISSIONS</td>
<td>NS88</td>
</tr>
</tbody>
</table>
Gams Data eXchange

• Gams Data eXchange (GDX):

• Complements the ASCII text data input

• Advantages:
  – Fast exchange of data
  – Syntactical check on data before model starts
  – Compile-time and Run-time Data Exchange
GDX Tools

- GDX Viewer
- gdxsplit
- gdxmerge
- gdxdump
- gdxdiff
- gdxxxrw (MS Office)
- gams
- GDX
- GDX API
- IDE
This email will submit an ICT model to the email submission tool.
ict@hillmodels.com  FINISHED 31517: Submitting an ICT model at ... 6/17/2002 5:01...
ict@hillmodels.com  STARTED 31517: Submitting an ICT model at ... 6/17/2002 4:59...
ict@hillmodels.com  SUBMITTED 31517: Submitting an ICT model a... 6/17/2002 4:58...
ict@hillmodels.com  NO XLS file in your mail 6/17/2002 4:42 PM
npm@hillmodels.com  DENIED: Testing access control 2 6/17/2002 4:31 PM
ict@hillmodels.com  DENIED: Testing access control 6/17/2002 4:26 PM
GRAS Architecture

Submission Tools
- Email: `submitemail.sh`
- Web: `save_file.pl`

Scheduler
- `sched.sh`

Workers
- `wsch.bat`
- A
- B
- C
- D
- E
Analytic Framework

- Optimization models do not allow for any type of vagueness
  - Input data requirements
  - Objectives and constraints
  - Results
- Misunderstandings result in failure of the model
  - Compilation/execution errors
  - Infeasible/unbounded MP models
- Model as a contract
Model as a Contract

- Good models do not rely on contract (input data)
- Input Module (handles bad data)
  - Simple error checks
  - Analyzing and reporting complex data problems
- Good models (modeling systems) provide access to results via independent *result analyzers* for non model experts
- Analytic framework help define *result metric*
  - e.g. violations of soft constraints
Scheduling US Military Academy West Point

“... each student’s daily activities are a carefully regimented balance of academic, military, and physical requirements.”
An Optimization Model

\[
\begin{align*}
\min & \sum_{ro} (p_{1,ro} \cdot \pi_{1,ro} + p_{2,ro} \cdot \pi_{2,ro}) + \sum_{c} (p_{3,c} \cdot \pi_{3,c} + p_{4,c} \cdot \pi_{4,c}) \\
\sum_{o} x_{c,ro} & = 1 \quad \text{(for all 8TAP entries)} \\
\sum_{r} x_{c,ro} & \leq 1 + \pi_{3,c} \quad \text{(for all cadets c for all time slots o)} \\
-\sigma - \pi_{4,c} & \leq \sum_{ro \text{ on day-1}} x_{c,ro} - \sum_{ro \text{ on day-2}} x_{c,ro} \leq \sigma + \pi_{4,c} \quad \text{(for all cadets c)} \\
x_{c,ro} & = 0 \quad \text{(for all c, ro where c has activity at o)} \\
\sum_{c} x_{c,ro} & \leq cap_{ro} + \pi_{1,ro} \quad \text{(for all course hours ro)} \\
\sum_{c \text{ freshman\&athlete}} x_{c,ro} - 0.6 \sum_{c} x_{c,ro} & \leq \pi_{2,ro} \quad \text{(for all course hours ro)}
\end{align*}
\]
Pre-Scheduling

- One cadet at a time
- Thousands of small MIPs
- If infeasible produce several infeasible schedules
- Human accepts infeasible schedule or modifies data
### Cadet Schedules with Constraint Violations, AYT 2001-1

**Header Information**

- **Select Constraint Type:** 3 FREE HOUR CONSTRAINT
- **Free Hour Violations:** 43
- **Design Group Violations:** 4
- **Unbalanced Schedule Violations:** 7

**Cadets With Schedule Violations**

<table>
<thead>
<tr>
<th>Course</th>
<th>Total Enrollment</th>
<th>Name</th>
<th>SSN</th>
<th>Grad Yr</th>
<th>Reviewed</th>
</tr>
</thead>
<tbody>
<tr>
<td>EM362A</td>
<td></td>
<td>BASS, WILLIE C.</td>
<td>158-</td>
<td>2002</td>
<td></td>
</tr>
<tr>
<td>PH365</td>
<td></td>
<td>BROWN, JAMEY A.</td>
<td>275-</td>
<td>2002</td>
<td></td>
</tr>
<tr>
<td>EM362A</td>
<td></td>
<td>BUNTING, BRIAN M.</td>
<td>220-</td>
<td>2002</td>
<td></td>
</tr>
<tr>
<td>EM301A</td>
<td></td>
<td>CHONOWSKI, DAVID P.</td>
<td>351-</td>
<td>2002</td>
<td></td>
</tr>
<tr>
<td>EN302</td>
<td></td>
<td>COOPER, GRAIG W.</td>
<td>242-</td>
<td>2002</td>
<td></td>
</tr>
<tr>
<td>EM301A</td>
<td></td>
<td>CULLUMBER, CRAIG M.</td>
<td>217-</td>
<td>2002</td>
<td></td>
</tr>
<tr>
<td>EM362A</td>
<td></td>
<td>DONNELL, TYLER R.</td>
<td>131-</td>
<td>2002</td>
<td></td>
</tr>
<tr>
<td>EM362A</td>
<td></td>
<td>EDGAR, BENJAMIN T.</td>
<td>411-</td>
<td>2002</td>
<td></td>
</tr>
</tbody>
</table>

**Details**

- **Name:** BASS, WILLIE C.
- **FOS1:** Civil Engineering Major
- **FOS2:**
- **Eng Seq Activity Code(s):** CIVIL ENGINEERING
- **CSWV:**
- **(3) 1 Day:**
  - TQPA: 2.414
  - CQPA: 2.699
- **(3) 2 Day:**

<table>
<thead>
<tr>
<th>Hour</th>
<th>Course</th>
<th>Violation</th>
<th>Override</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>PE310</td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>MA364</td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>PL300</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>PL300</td>
<td></td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>EM362A</td>
<td>FREE HOUR CONSTRAINT</td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>EM362A</td>
<td>FREE HOUR CONSTRAINT</td>
<td></td>
</tr>
</tbody>
</table>

**Z Hour**

<table>
<thead>
<tr>
<th>Hour</th>
<th>Course</th>
<th>Violation</th>
<th>Override</th>
</tr>
</thead>
<tbody>
<tr>
<td>G</td>
<td>SS307</td>
<td></td>
<td></td>
</tr>
<tr>
<td>H</td>
<td>HI301</td>
<td></td>
<td></td>
</tr>
<tr>
<td>I</td>
<td>EM364A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>J</td>
<td>EM364A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>K</td>
<td>,R</td>
<td></td>
<td></td>
</tr>
<tr>
<td>L</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Schedule**

- **OK**
- **Close**
## Results

- AY 2000/2 parallel tested
- AY 2001/1 deployed

<table>
<thead>
<tr>
<th></th>
<th>Legacy System + human deconflicter</th>
<th>New System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Individual Relaxations</td>
<td>203/304/116</td>
<td>58/25/4</td>
</tr>
<tr>
<td>Capacity Overloads</td>
<td>12/54</td>
<td>9/21</td>
</tr>
<tr>
<td>Number of Schedulers</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>Time to produce Schedule</td>
<td>4 Weeks</td>
<td>1 Day</td>
</tr>
</tbody>
</table>
Cost Saver

• Most convincing and obvious reason for using an optimization model
• *Science of better (INFORMS)*
• Often exaggerated/difficult to estimate
• More reasons:
  – Institutionalize personal knowledge
  – Scientific foundation (economic models)
  – Get “fair” results (usually fails)
Model Roles over Time

Communication Vehicle

Analytic Framework

Cost Saver

Lifecycle: +15 Years
Long Term Commitment

• Backward compatibility
• New Solvers/Platforms
• Performance comparison tools: Bench / Paver
• Model converter and “encryption“ tool: Convert
• Software Quality Assurance (SQA)
  – Software configuration management
  – Quality control and tests of the product
  – Client model testing
Quality Control and Tests of the Product

• Goal: Continuous quality improvement using automated and reproducible tests
• Test libraries (available online):
  – GAMS Model Library
  – GAMS Quality Test Models Library
  – Solved for all relevant solvers: More than 16,000 solves for each platform
• Quality Test Models Library
• Include tests to verify proper behavior of the system
• More than 140 quality test models, each containing numerous pass/fail tests:
  \texttt{abort$\text{card}(\text{delta}) \ 'time \ routines \ have \ an \ error';}
• Automatic generated test summaries with different level of information
Summary of two quality runs

*** Status: Normal completion
--- quality.gms(284) 4 Mb
--- quality.gms(287) 4 Mb 1 Error
There were errors: 4 out of 267 tests failed.
See the file failures.gms to reproduce the failed runs
--- Putfile this D:\support\testlib\onetest.gms
--- quality.gms(287) 4 Mb 1 Error
*** Status: Execution error(s)

================================================

*** Status: Normal completion
--- quality.gms(284) 4 Mb
--- quality.gms(295) 4 Mb
Congratulations! All 267 tests passed.
See the file alltests.gms to reproduce all the runs
--- Putfile this D:\support\testlib\onetest.gms
*** Status: Normal completion
Client Model Testing

- Client with complex application (energy management system)
- New GAMS version available:
  - Relevant new features?
  - Performance gains?
  - No „surprises“?
    - Bugs
    - Different results (e.g. MIP models)
"After upgrading GAMS on our machines to the latest distribution, runs take about twice or three times as much time as before (3 to 4 hours instead of 1 or 1 and half). We decided to downgrade and investigate the problem later."
• Want guarantee that their application will work with the new version
• Only limited resources to do major testing themselves
• Confidentiality issues: Running tests without having access to internal model structures and model data (in a human readable format)
Client Model Testing

- Requires changes to the model of the clients to allow automated pass/failure tests
- Gives clients assurance that their application will also work with new GAMS releases
- Includes:
  - Ability to solve (= no bugs)
  - Returns the same solution back
  - Similar or better performance
- Improves communication between development team and clients (specific wishes)
Conclusions

- Model can contribute to a project at various stages
- Although often small in budget, the modeling tasks can become the central core in a project
- Long term commitments in various areas are necessary, new challenges in client model testing.