Deploying Your Application Built Around GAMS

Lutz Westermann

GAMS Software GmbH
GAMS Development Corporation

lwestermann@gams.com

www.gams.com
GAMS at a Glance

General Algebraic Modeling System

- Algebraic Modeling Language
- 30+ Integrated Solvers
- 10+ Supported MP classes
- 10+ Supported Platforms
- Connectivity & Productivity Tools
  - IDE
  - Model Libraries
  - GDX, Interfaces & Tools
  - Grid Computing
  - Benchmarking
  - Compression & Encryption
  - Deployment System
  - APIs (C, Fortran, Java, .Net ...)
  - ...

...
Outline

• Prerequisites
  – Object Oriented GAMS API
  – Cutting Stock Problem

• Building an Application around GAMS

• What’s New
Outline

• Prerequisites
  – Object Oriented GAMS API
  – Cutting Stock Problem

• Building an Application around GAMS

• What’s New
Calling GAMS from your Application

Creating Input for GAMS Model
→ Data handling using GDX API

Callout to GAMS
→ GAMS option settings using Option API
→ Starting GAMS using GAMS API

Reading Solution from GAMS Model
→ Data handling using GDX API
Low level APIs → Object Oriented API

- Low level APIs
  - GDX, OPT, GAMSX, GMO, …
  - High performance and flexibility
  - Automatically generated imperative APIs for several languages (C, Delphi, Java, Python, C#, …)

- Object Oriented GAMS API
  - Additional layer on top of the low level APIs
  - Object Oriented
  - Written by hand to meet the specific requirements of different Object Oriented languages
Features of the object oriented API

- No modeling capability, model is still written in GAMS

- Prepare input data and retrieve results in a convenient way → `GAMSDatabase`

- Control GAMS execution → `GAMSJob`

- Scenario Solving: Feature to solve multiple very similar models in a dynamic and efficient way. → `GAMSMModelInstance`

- Seamless integration of GAMS into other programming environments
Outline

• Prerequisites
  – Object Oriented GAMS API
  – Cutting Stock Problem

• Building an Application around GAMS

• What’s New
Cutting Stock Problem: Objective

• Cut out paper products …
  – of different sizes
  – from large raw paper rolls
  – in order to meet customer's demand

• Objective: minimize the required number of paper rolls
Cutting Stock Problem: Input / Output

• Input:
  – Number of different products with…
    • Width of product
    • Demand of product
  – Width of raw paper roll

• Output:
  – Different cutting patterns
  – Usage of those patterns
    ➔ Number of required raw paper rolls
Cutting Stock Problem: Column Generation

Master Model → Demand Duals → Pricing Model

New Patterns → cutstock.gms
Outline

• Prerequisites
  – Object Oriented GAMS API
  – Cutting Stock Problem

• Building an Application around GAMS

• What’s New
Application Step 1

- What are we going to do?
  1. Run the model
  2. Print out the results
  3. Change the solver used

- Interface:

- What do we need?
  1. GAMSWorkspace & GAMSJob
  2. GAMSPParameter
  3. GAMSOptions (Solvers)
Application Step 2

- **What are we going to do?**
  - Define the input data within the application
  - Store the data and pass it to the model
  - Check for Errors

- **Interface:**

- **What do we need?**
  - GAMSDatabase
  - GAMSOptions (Defines)
  - GAMSException

\[ i, r, w(i), d(i) \]
\[ \text{patrep}(*,*) \]
Application Step 3

• What are we going to do?
  – Move the logic of the algorithm from GAMS into the application layer (improve performance)

• Interface:
  C#

• What do we need?
  – GAMSCheckpoint
  – GAMSModelInstance
  – GAMSModifier

C# to GAMS

i, r, w(i), d(i)

GAMS Master

Demand.m

GAMS Pricing

xp(p), aip(i, p)
Excursus: GAMSModelInstance etc.

**GAMSJob**
- Manages the execution of a GAMS program given by GAMS model source

**GAMSCheckpoint**
- Captures the state of a GAMSJob

**GAMSModelInstance**
- A single mathematical model generated by a GAMS solve statement

**GAMSModifier**
- Marks elements of a GAMSModelInstance to be modifiable

→ Step3.cs
Application Step 4

• What are we going to do?
  – Add a graphical user interface

Step4.cs
Deploying the Application

![GAMS Component Wizard](image)
Outline

• Prerequisites
  – Object Oriented GAMS API
  – Cutting Stock Problem

• Building an Application around GAMS

• What’s New
New GAMS Distribution 23.9.3

Released September, 27th

- Solver updates
  - BARON 11.3
  - CONOPT 3.15F
  - CPLEX 12.4 fixpack 1
  - GLOMIQO 2.0
  - GUROBI 5.0.1
  - KNITRO 8.0
  - LINDO 7.0.1.487
  - MOSEK 6 rev 137
  - XPRESS 23.0.05
  - …

www.gams.com/download
Object Oriented GAMS API

- Object Oriented API provides an additional abstraction layer of the low level GAMS APIs
- Powerful and convenient link to other programming languages
- .NET API with many examples is part of the current GAMS release available at www.gams.com
- Python and Java under development
GDXRRW

- GDXRRW bridges the gap between R and GAMS
- Fits into the ecosystem of existing GDX utilities
- Presents data in a natural form for R users

Source: http://blog.modelworks.ch

GDXRRW: Exchanging Data between GAMS and R, Steven Dirkse, Monday, 11am, Suite 322
Stochastic Programming in GAMS

• 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)
Contacting GAMS

**Europe**

GAMS Software GmbH  
P.O. Box 40 59  
50216 Frechen,  
Germany

Phone: +49 221 949 9170  
Fax: +49 221 949 9171  

info@gams.de

**USA**

GAMS Development Corp.  
1217 Potomac Street, NW  
Washington, DC 20007  
USA

Phone: +1 202 342 0180  
Fax: +1 202 342 0181  

sales@gams.com  
support@gams.com

http://www.gams.com