

**GAMS**



# Rapid Application Prototyping With GAMS

Lutz Westermann

[lwestermann@gams.com](mailto:lwestermann@gams.com)

GAMS Development Corporation

[www.gams.com](http://www.gams.com)

GAMS Software GmbH

[www.gams.de](http://www.gams.de)



**INFORMS 2009 San Diego**



# GAMS at a Glance

The screenshot displays the GAMS software interface with several components:

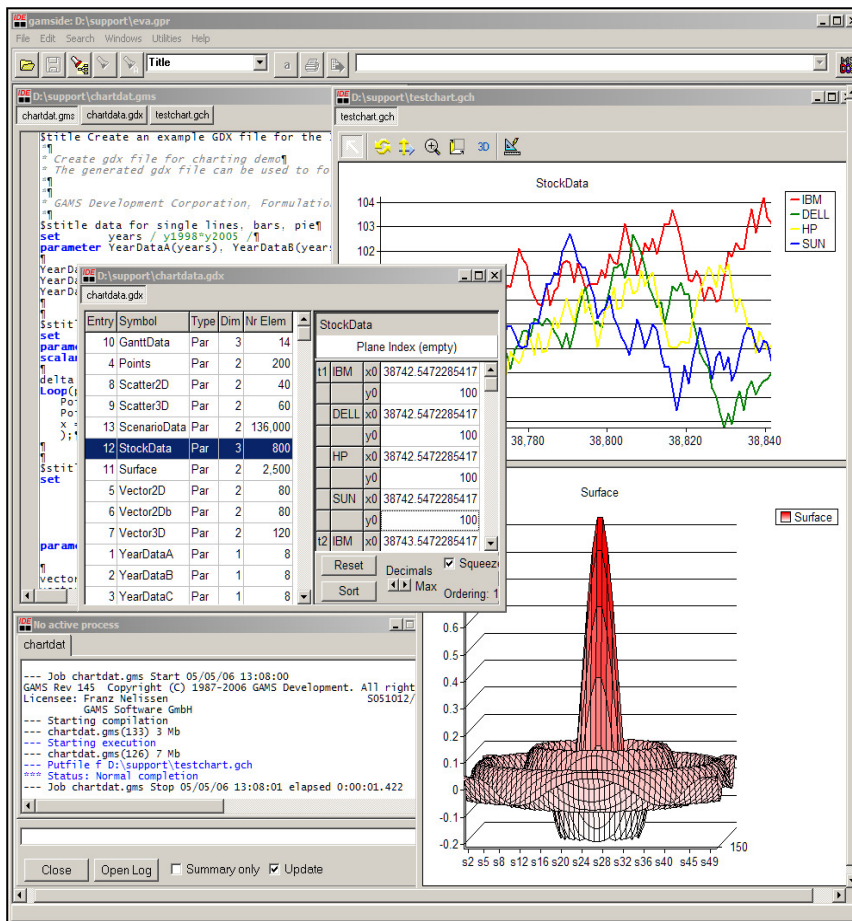
- Code Editor:** Contains GAMS code for creating an example GDY file for charting. The code includes comments and parameters for data sets.
- Data Table:** A table listing model elements with columns for Entry, Symbol, Type, Dim, and Nr Elem. The 'StockData' entry is highlighted.
- StockData Chart:** A line chart showing the stock prices of IBM, DELL, HP, and SUN over time. The y-axis ranges from 102 to 104, and the x-axis ranges from 38,780 to 38,840.
- Surface Chart:** A 3D surface plot showing a sharp peak. The y-axis ranges from -0.2 to 0.6, and the x-axis ranges from s2 to s49.
- Log Window:** Shows the execution status of the job, including start and stop times and elapsed time.

## Algebraic Modeling System

- Facilitates to formulate mathematical optimization problems similar to algebraic notation
  - ➔ Simplified model building
- Provides links to appropriate state-of-the-art external algorithms
  - ➔ Efficient solution process

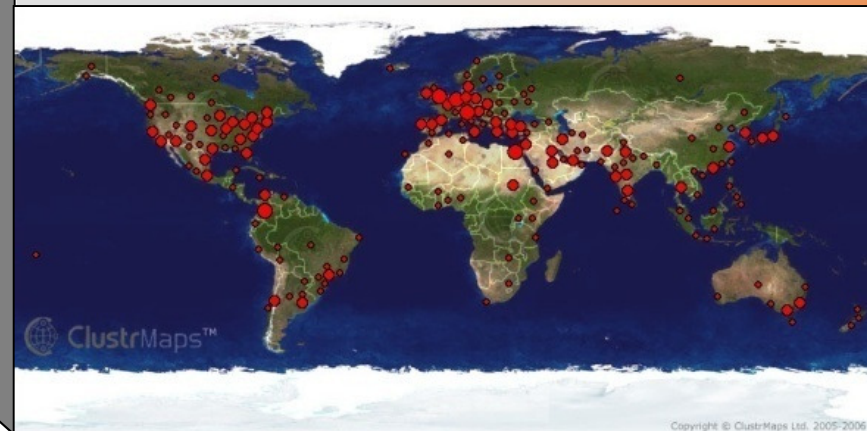


# GAMS at a Glance



## General Algebraic Modeling System

- Roots: World Bank, 1976
- Went commercial in 1987
- GAMS Development Corp.
- GAMS Software GmbH
- Broad academic & commercial user community and network







# GAMS at a Glance

The screenshot displays the GAMS IDE with several windows:

- Code Editor:** Contains GAMS code for creating a GDX file and defining data for a charting demo. It includes comments and parameter declarations like `parameter YearDataA(years), YearDataB(years);`.
- Data Table:** A table listing model elements:
 

Entry	Symbol	Type	Dim	Nr Elem
10	GanttData	Par	3	14
4	Points	Par	2	200
8	Scatter2D	Par	2	40
9	Scatter3D	Par	2	60
13	ScenarioData	Par	2	136,000
12	StockData	Par	3	800
11	Surface	Par	2	2,500
5	Vector2D	Par	2	80
6	Vector2Db	Par	2	80
7	Vector3D	Par	2	120
1	YearDataA	Par	1	8
2	YearDataB	Par	1	8
3	YearDataC	Par	1	8
- StockData Plot:** A line graph showing stock price fluctuations for IBM (red), DELL (green), HP (yellow), and SUN (blue) over time.
- Surface Plot:** A 3D surface plot showing a sharp peak, with axes labeled s2, s5, s8, s12, s16, s20, s24, s28, s32, s36, s40, s45, s49.
- Log Window:** Shows the execution status of the job, including start and stop times and elapsed time.

## General Algebraic Modeling System

- Algebraic Modeling Language
- 25+ 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
  - ...

# GAMS

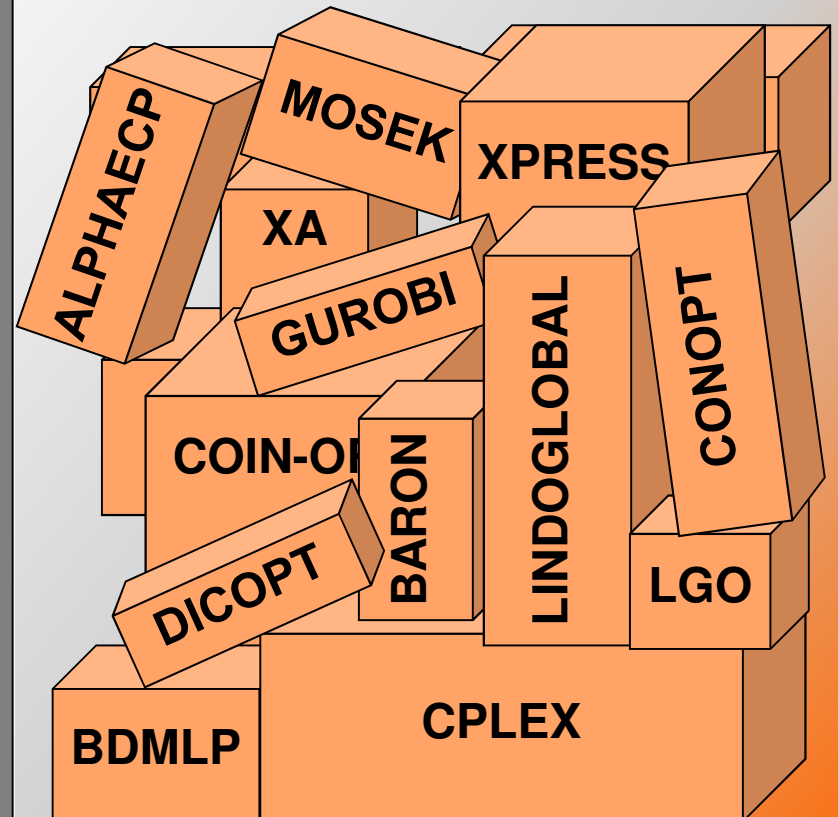


## GAMS at a Glance

The screenshot shows the GAMS software interface with the following components:

- Code Editor:** Contains GAMS code for creating a GDX file and defining data for a charting demo. The code includes comments and commands like `set`, `parameter`, `YearDataA`, and `YearDataB`.
- Data Table:** A table listing model elements with columns for Entry, Symbol, Type, Dim, and Nr Elem. The selected entry is 12 StockData, which has 3 dimensions and 800 elements.
- StockData Plot:** A line graph showing the values of four stocks (IBM, DELL, HP, SUN) over time. The y-axis ranges from 102 to 104, and the x-axis ranges from 38,780 to 38,840.
- Surface Plot:** A 3D surface plot showing a sharp peak. The y-axis ranges from -0.2 to 0.6, and the x-axis ranges from s2 to s49.
- Log Window:** Shows the execution log for the job `chartdat.gms`, indicating that the job completed normally on 05/05/06 at 13:08:01, with an elapsed time of 0:00:01.422.

25+ Integrated Solvers



# GAMS



## GAMS at a Glance

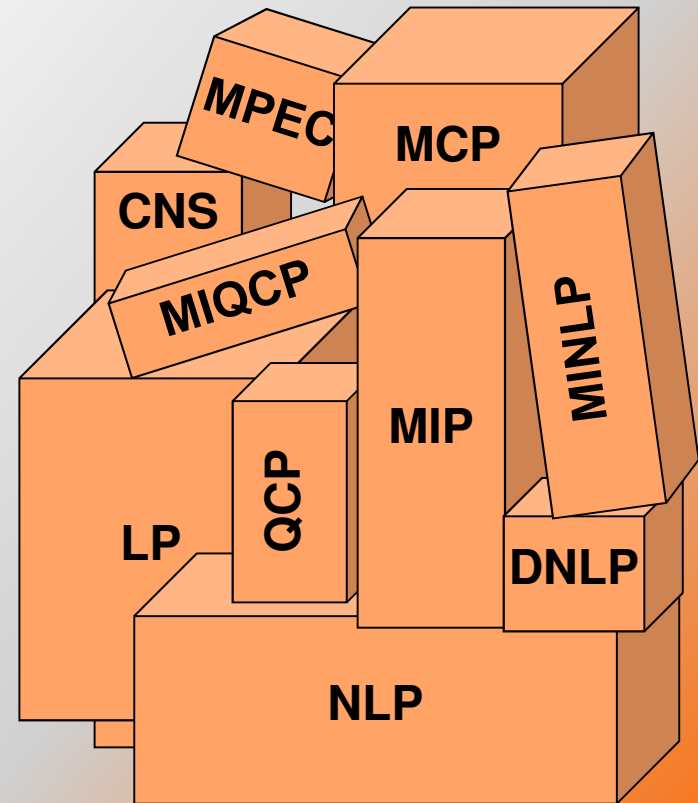
The screenshot displays the GAMS software interface with several windows open:

- Code Editor:** Shows GAMS code for creating an example GDY file for charting. The code includes comments and commands like `set years y1998:y2005 /;` and `parameter YearDataA(years), YearDataB(years);`.
- Table:** A table listing model elements. The selected row is:

Entry	Symbol	Type	Dim	Nr Elem
10	GanttData	Par	3	14
4	Points	Par	2	200
8	Scatter2D	Par	2	40
9	Scatter3D	Par	2	60
13	ScenarioData	Par	2	136,000
12	StockData	Par	3	800
11	Surface	Par	2	2,500
5	Vector2D	Par	2	80
6	Vector2Db	Par	2	80
7	Vector3D	Par	2	120
1	YearDataA	Par	1	8
2	YearDataB	Par	1	8
3	YearDataC	Par	1	8

- StockData Plot:** A line graph showing stock prices for IBM (red), DELL (green), HP (yellow), and SUN (blue) from 1998 to 2005. The y-axis ranges from 102 to 104.
- Surface Plot:** A 3D surface plot showing a sharp peak. The x-axis is labeled with values s2, s5, s8, s12, s16, s20, s24, s28, s32, s36, s40, s45, s49. The z-axis ranges from -0.2 to 0.6.
- Log Window:** Shows the execution log for the job `chartdat.gms`, indicating a normal completion.

10+ Supported MP classes





# GAMS



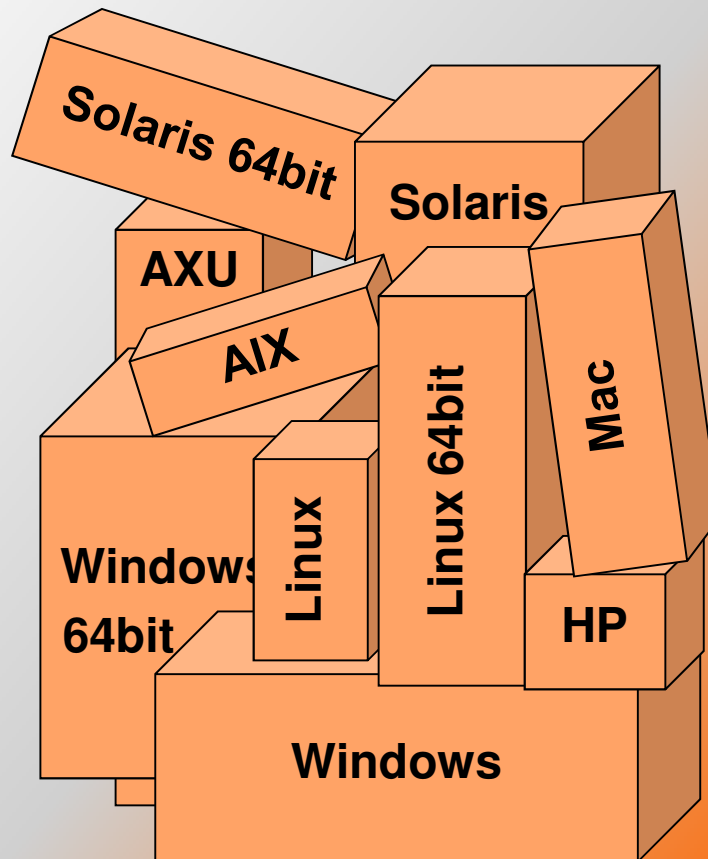
## GAMS at a Glance

The screenshot displays the GAMS development environment. The main window shows GAMS code for creating a GDX file and plotting data. A 'StockData' window shows a line chart with four series: IBM (red), DELL (green), HP (yellow), and SUN (blue). A 'Surface' window shows a 3D wireframe plot of a surface. A table window lists the data elements:

Entry	Symbol	Type	Dim	Nr Elem
10	GanttData	Par	3	14
4	Points	Par	2	200
8	Scatter2D	Par	2	40
9	Scatter3D	Par	2	60
13	ScenarioData	Par	2	136,000
12	StockData	Par	3	800
11	Surface	Par	2	2,500
5	Vector2D	Par	2	80
6	Vector2Db	Par	2	80
7	Vector3D	Par	2	120
1	YearDataA	Par	1	8
2	YearDataB	Par	1	8
3	YearDataC	Par	1	8

At the bottom, a log window shows the execution status: 'Job chartdat.gms Start 05/05/06 13:08:00', 'Starting compilation', 'Starting execution', and 'Status: Normal completion'.

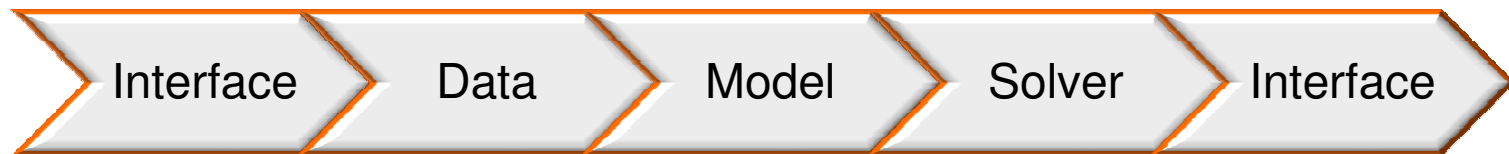
10+ Supported Platforms





# GAMS' Fundamental concepts

- **Platform independence**
- **Open architecture and interfaces to other systems**
- **Balanced mix of declarative and procedural elements**
  - Declaration of Sets, Parameters, Variables, Equations, Models, ...
  - Procedural Elements like loops, if-then-else, ...
- **Layers of separation**







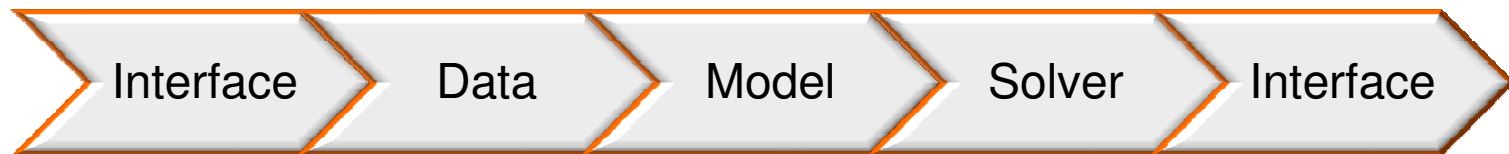
# GAMS' Fundamental concepts

- **Different layers with separation of**

- model and data
- model and solution methods
- model and operating system
- model and interface

→ **Models benefit from**

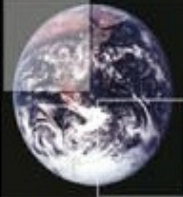
- advancing hardware
- enhanced / new solver technology
- improved / upcoming interfaces to other systems





## GAMS 23.3 Beta

- Released yesterday! [www.gams.com/beta](http://www.gams.com/beta)
- Solver updates:
  - Baron 9 (Conopt as an NLP solver)
  - Gurobi 2.0
  - Mosek 6 (beta)
  - Xpress 20.00
  - Coin-OR (various)
  - Coin-OR based Cplex, Gurobi, Mosek, Xpress links
- GAMS on Amazon EC2 (pay by the hour)



# Basic Sudoku

Address  <http://www.dailysudoku.com/sudoku/index.shtml>

## Daily SuDoku



Home

Today's SuDoku

SuDoku Archive

SuDoku for Kids

Draw/Play

Discussion

FAQ

Books

Syndication

Links

Email and News

Contact

### 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 Sudokus** are exactly the same, but more numbers and letters are needed.

Classic

Monster

Kids

Squiggly

					6	1
			7	3	1	4
5				9		
6			2			1
		8			4	
	1			5		8
			9			3
7			8	6	3	
9		2				

Daily SuDoku: Thu 2-Nov-2006

very hard

(c) Daily Sudoku Ltd 2006. All rights reserved.





# Christmas Tree Sudoku

Address <http://www.dailysudoku.com/sudoku/archive.shtml?year=2005&month=12&day=23&type=seasonal>

## Daily SuDoku



- Home
- Today's SuDoku
- SuDoku Archive**
- SuDoku for Kids
- Draw/Play
- Discussion
- FAQ
- Books
- Syndication
- Links
- Email and News
- Contact

Daily Seasonal Sudoku: Fri 23-Dec-2005 [\[instructions\]](#)

	3			2			9
		1				2	
			7		3		
	7		4		9		2
	6	2				8	3
			1		5		
			8		4		
3							5

© Daily Sudoku Ltd 2005. All rights reserved.

Christmas tree Sudoku: Fri 23-Dec-2005 very hard



## Demo: Basic Sudoku (*su1*)

Basic model  
*su1* computes  
solution to  
given Sudoku

```

IDE File Edit Search Windows Utilities Model Libraries Help
[Icons] [a] [Print] [Run] gdx=su1
su1.gms
$title SUDOKU model 1

* Define SUDOKU grid and basic relationships
Sets  r  rows      / r1*r9 /
      c  columns   / c1*c9 /
      b  blocks    / b1*b9 /
      v  values    / v1*v9 /
br(b,r) / b1*b3 .r1*r3, b4*b6 .r4*r6, b7*b9 .r7*r9 /
bc(b,c) / (b1,b4,b7).c1*c3, (b2,b5,b8).c4*c6, (b3,b6,b9).c7*c9 /
brc(b,r,c) block definitions ;

brc(b,r,c) = br(b,r) *bc(b,c) ;

Table problem(r,c) Hard problem with non-unique solution
      c1  c2  c3  c4  c5  c6  c7  c8  c9
r1    2           6  7
r2           6           2
r3    4           8           1
r4    5           9  3
r5           3           5
r6           2  8           7
r7           1
r8    7           8           6
r9           5  3           8 ;

```



## Demo: Find other solutions ( $su1 \rightarrow su2$ )

- Is the solution unique?
- If not, how many solutions exist?
- Edits for  $su1 \rightarrow su2$ :
  - Implement binary cuts to exclude known solution
  - Use loop to find and store solutions





## Demo: Infeasible Sudoku ( $su1 \rightarrow su3$ )

- 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  $su1 \rightarrow su3$ 
  - Use random generation to get bogus data
  - Remove X.fx for fixed cells
  - Add binary variable UNDO (relaxes fixed cells)
  - Add equation fix using the UNDO variables
  - Add new objective function: Minimize sum over all UNDOs
  - Write short report







GAMS

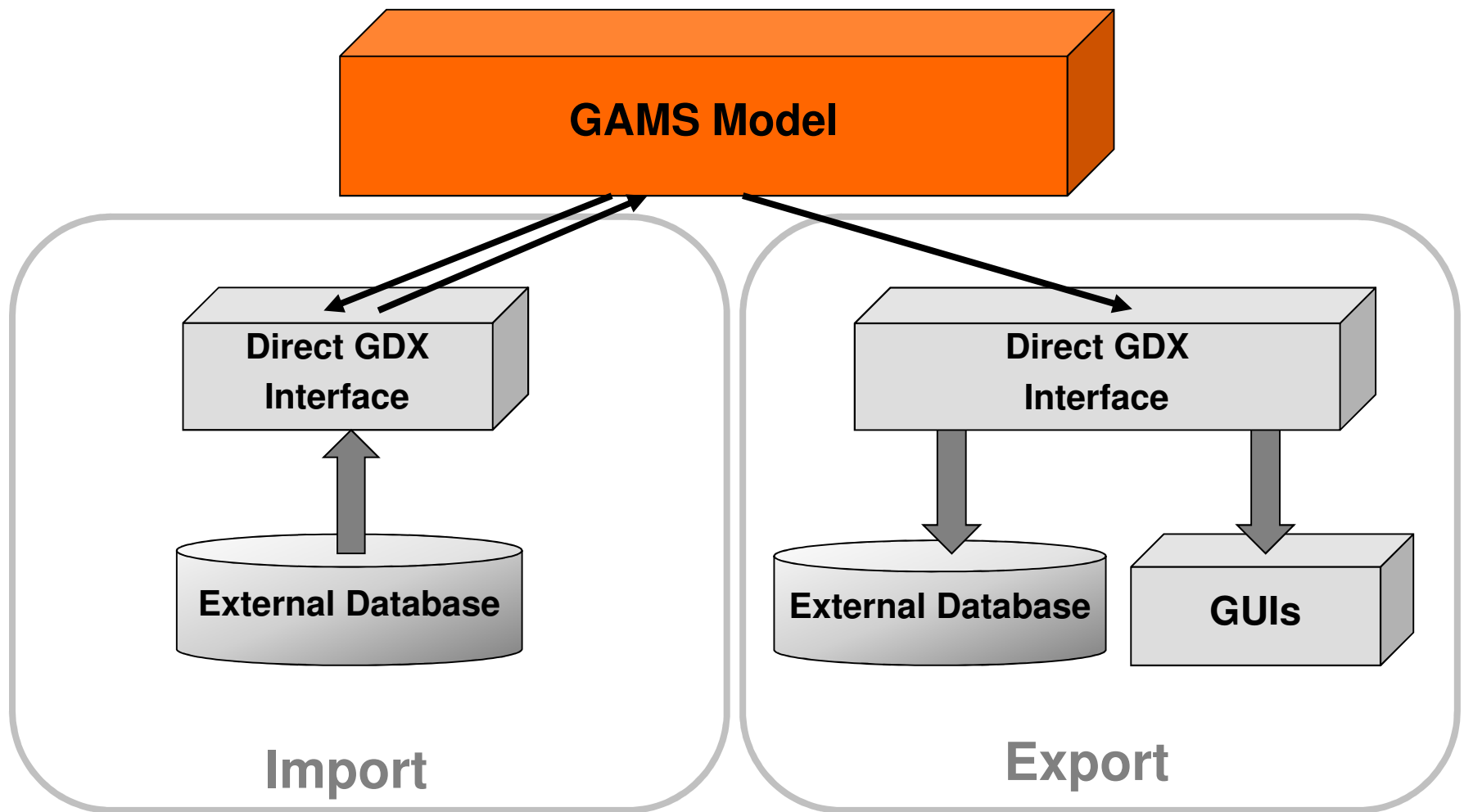


## Demo: Samurai model (su3 → su4)

- 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

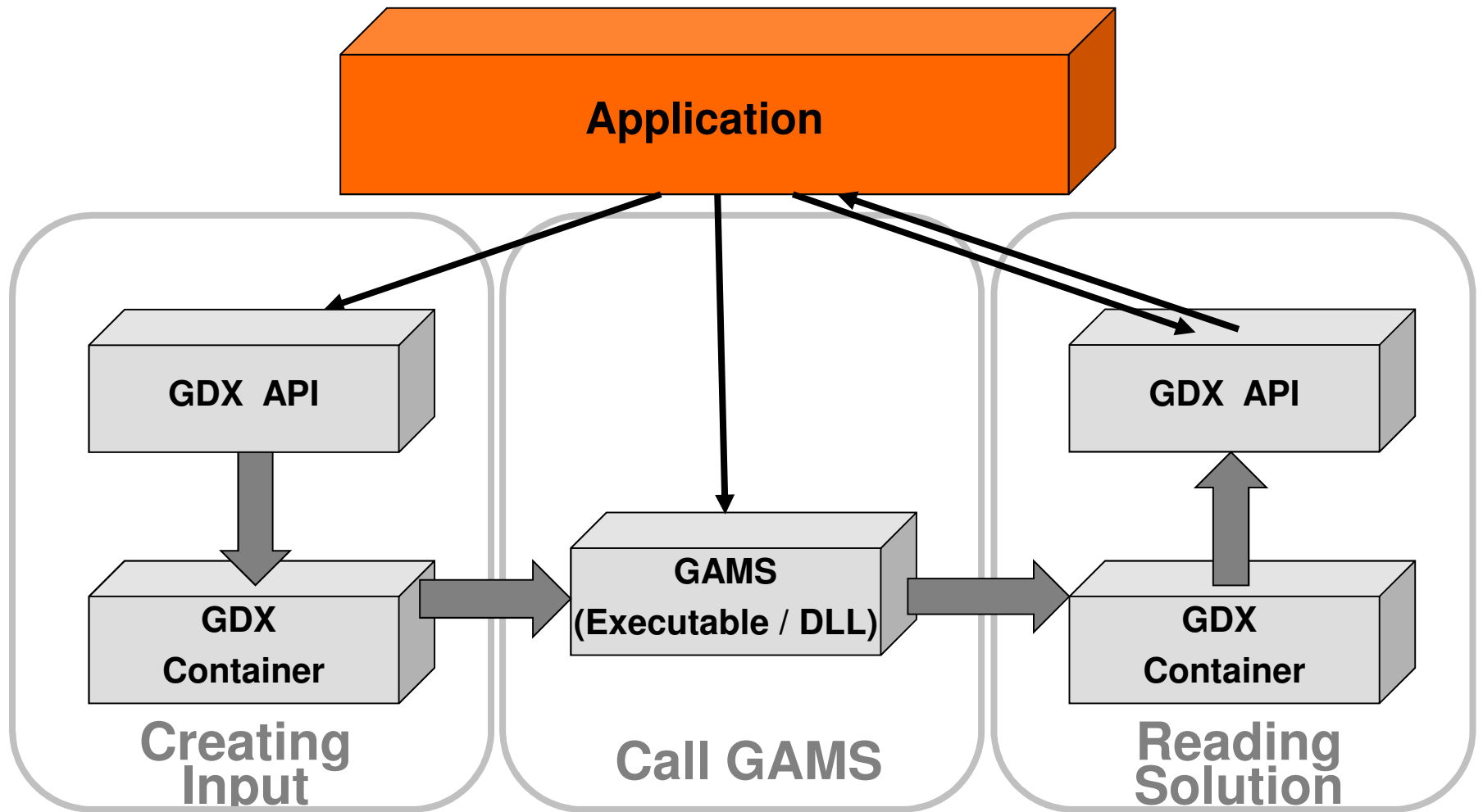


# GAMS in Control





# Application in Control



**GAMS**



# Calling GAMS from an Application

**Creating Input for GAMS Model**

**Callout to a GAMS Process/Executable**

**Reading Output from GAMS Model**

- Works from basically every environment
  - Web application (server side)
  - Application Builder
    - Oracle, Eclipse, .NET, ...
    - Regular Programming language C++, Java, VB, ...
  - MS Office Application / VBA
- Integrates with existing user IT infrastructure





# Demo: Excel in charge (samurai\_vb)

- Existing Samurai model with Excel GUI
- Look at data communication between model and GUI

8				6			2				3			
4	7				5	1	3	7			8	9		
	5	8		1	9			5	7		3	1		
	9		5	6				9		5	4			
		1		6					8		9			
	3		7		2			4	7		3			
	4	5		3	1			2	4		7	6		
5	3				2	4		1	3			7	4	
6					3		2	4					2	
							4		6					
					3					8				
						3		8						
5				7		1		8					1	
2	4			9	1			7	2			9	8	
	9	8		2	6				9	7		6	2	
	2		4	9					3	2		4		
		2		7						6		3		
	1		3		2				7		1	5		
	8	5		3	4				2	4		9	1	
6	1				5	8		1	7				4	5
2					3			4						2

Samurai 1	Samurai 2
Samurai 3	Samurai 4
<b>Solve</b>	

GAMS Directory:	c:\program files\gams23.2\
Working Directory:	c:\tmp3\

GAMS



## Demo: Samurai data input ( $su4 \rightarrow su5$ )

- Prepare our Samurai model  $su4$  to plug in to spreadsheet
- Import 21x21 data from GUI (via GDX)
- Use mappings from map1 to map 21x21  $\rightarrow$  5x9x9
- Export 21x21 solution to GDX



## Demo: Clean up (su5 → su6)

- Create text file for display in GUI

```
Solver: CPLEX  
equations: 1945 variables: 3646  
model status: 1 OPTIMAL  
solver status: 1 NORMAL COMPLETION  
iterations: 0 solve time: 0.08
```

**GAMS**



# 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>

[info@gams.de](mailto:info@gams.de)

[support@gams-software.com](mailto:support@gams-software.com)

## 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](mailto:sales@gams.com)

[support@gams.com](mailto:support@gams.com)