



Recent enhancements

in **GAMS**

Lutz Westermann

lwestermann@gams.com

GAMS Software GmbH

GAMS Development Corporation

www.gams.com



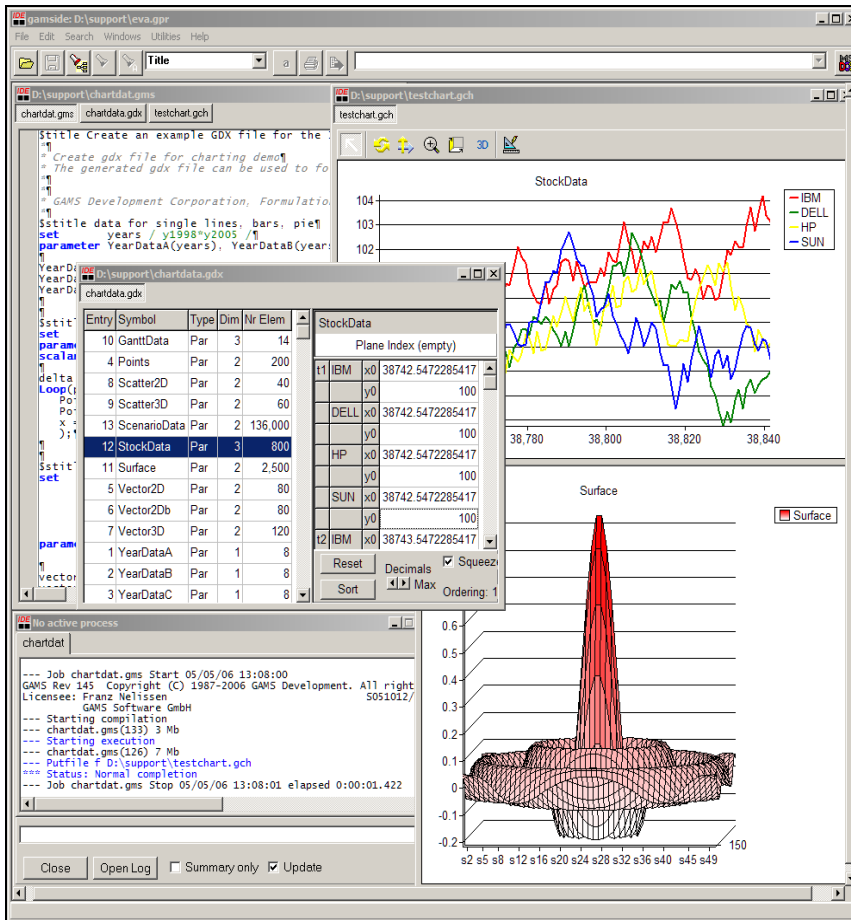
International Conference on Operations Research

MUNICH 2010

September 1st - 3rd



GAMS at a Glance

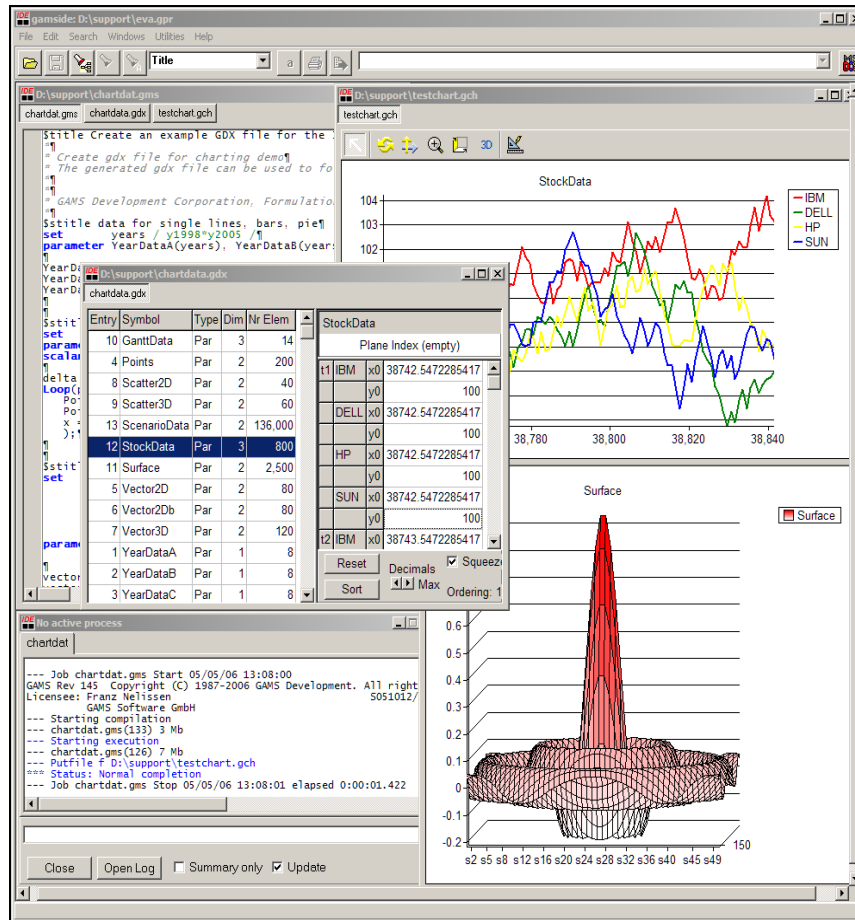


Algebraic Modeling System

- 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

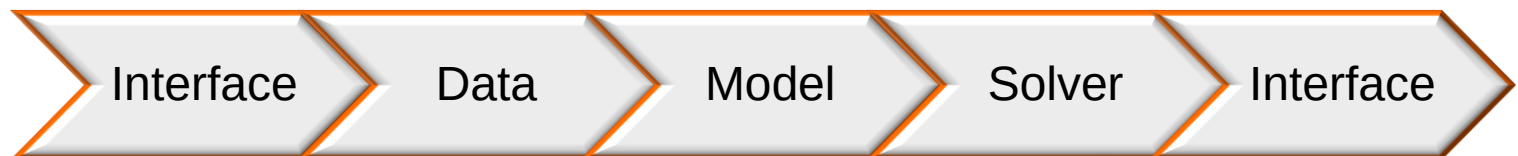
- Roots: World Bank, 1976
- Went commercial in 1987
- GAMS Development Corp.
- GAMS Software GmbH
- Broad academic & commercial user community and network





GAMS' 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, ...





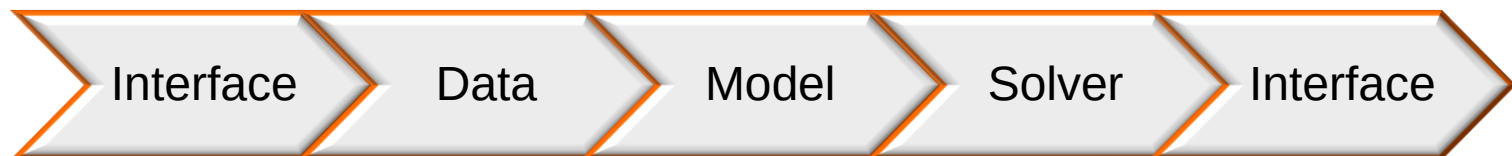
GAMS' Fundamental concepts

- **Different layers with separation of**

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

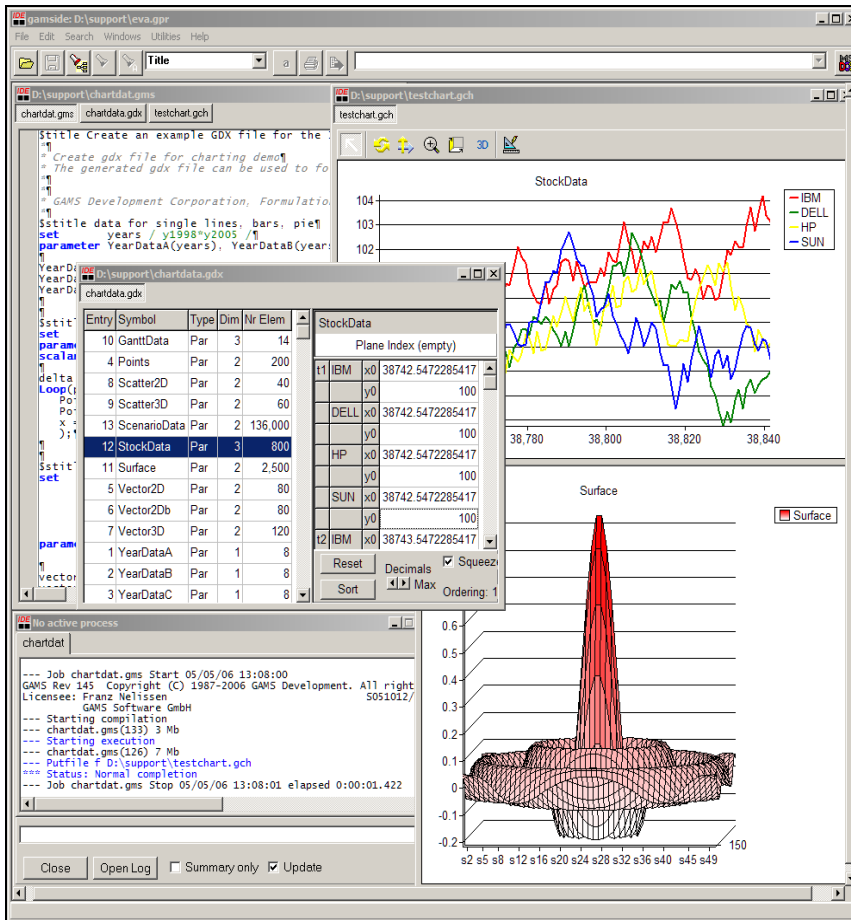
→ **Models benefit from**

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





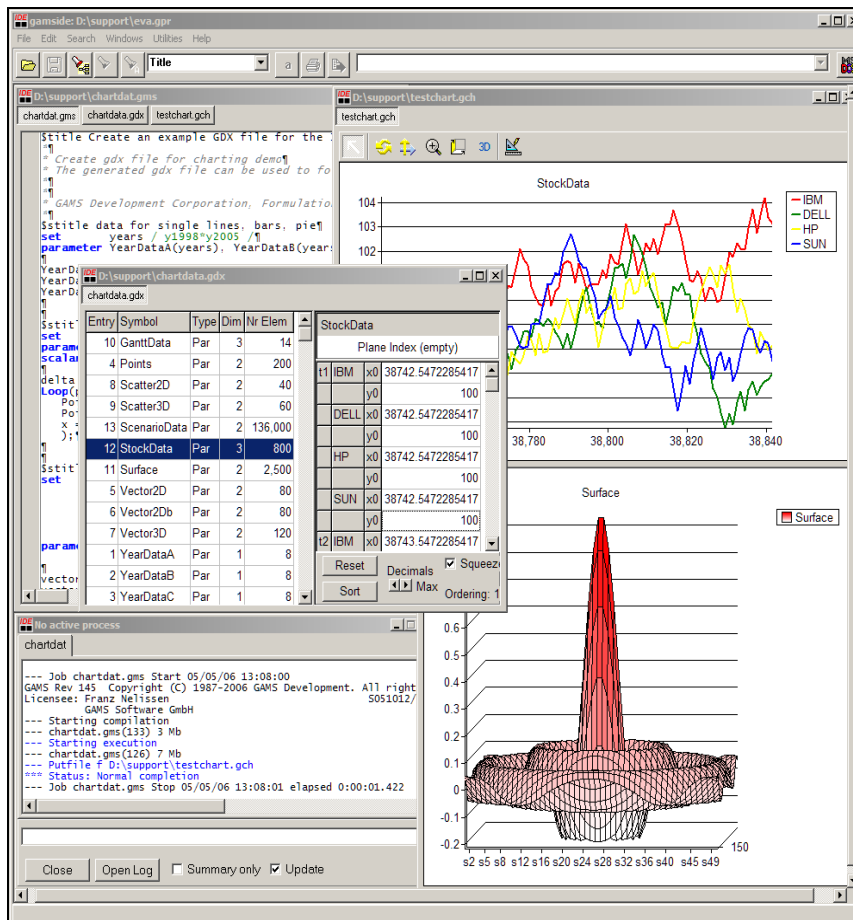
GAMS at a Glance



General Algebraic Modeling System

- Algebraic Modeling Language
- 10+ Supported Platforms
- 25+ Integrated Solvers
- 10+ Supported MP classes
- Connectivity- & Productivity Tools
 - IDE
 - Model Libraries
 - GDX, Interfaces & Tools
 - Grid Computing
 - Benchmarking
 - Compression & Encryption
 - Deployment System
 - ...

GAMS at a Glance



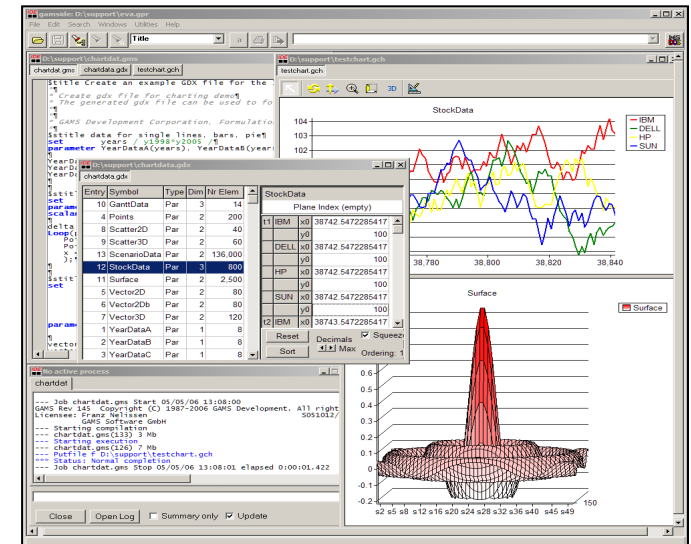
The GAMS/BASE Module

- Compiler and Execution System
- GAMS IDE (Windows)
- Documentation + Model libraries
- GDX Utilities
- Free Solvers/Solver Links



Integrated Development Environment

- 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





Documentation

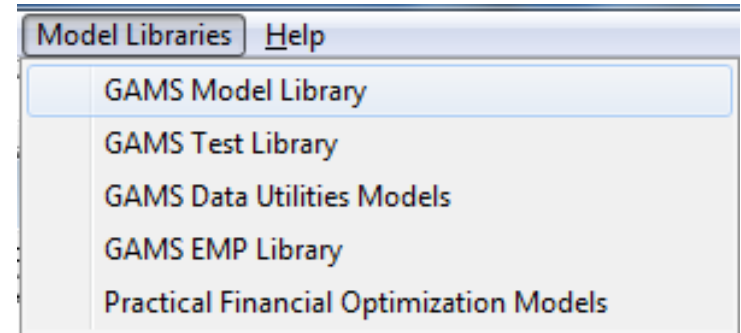
- **Distributed Documentation**
 - GAMS Users Guide
 - Expanded GAMS Users Guide (McCarl)
 - Solver Manuals
 - GAMS Utility Manuals
- **Wikis**
 - Support Wiki <http://support.gams-software.com>
 - Interfaces Wiki <http://interfaces.gams-software.com>
- **Search all GAMS Websites**
<http://www.gams.com/search.htm>



Distributed Model Libraries

- **GAMS Model Library**

- Example and user-contributed models
- Very often used as templates
- Tests for
 - Solver robustness and correctness
 - Backward compatibility



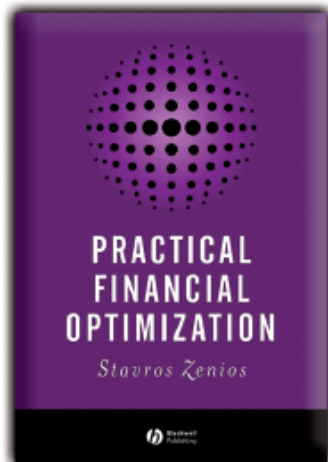
- **GAMS Test Library**

- Transparent and reproducible Quality Assurance Tests
- Tests for
 - Solver correctness
 - Special functions
 - GAMS utilities



Distributed Model Libraries

- **GAMS Data Utilities Library**
 - Demonstration of the various utilities interfacing GAMS with other applications
 - E.g. gdxxrw, mdb2gms, sql2gms
- **GAMS EMP Library**
 - Examples for the use of Extended Mathematical Programming



- **Practical Financial Optimization Models**
Models of the book

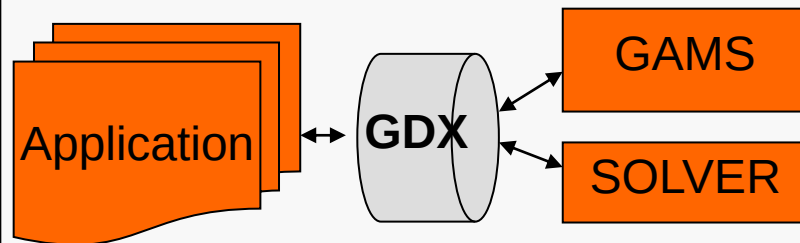
*“PRACTICAL FINANCIAL OPTIMIZATION –
A Library of GAMS Models”*

by Consiglio, Nielsen and Zenios



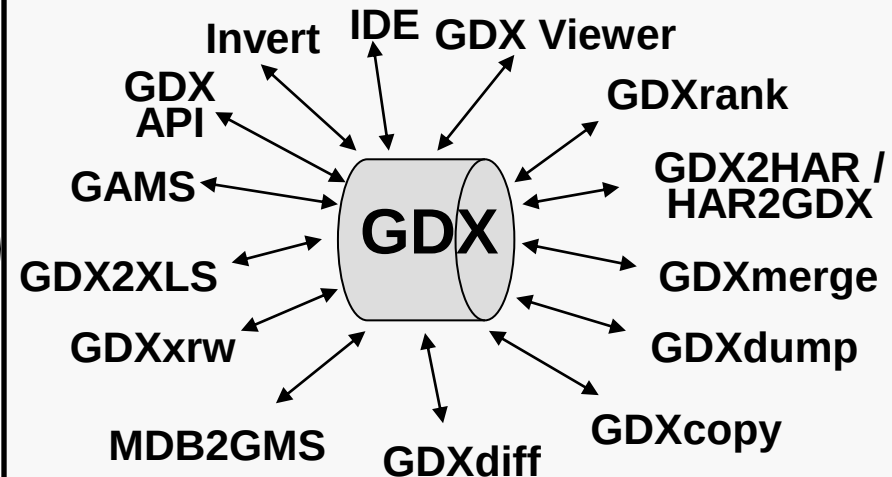
Gams Data eXchange

Binary Data Exchange



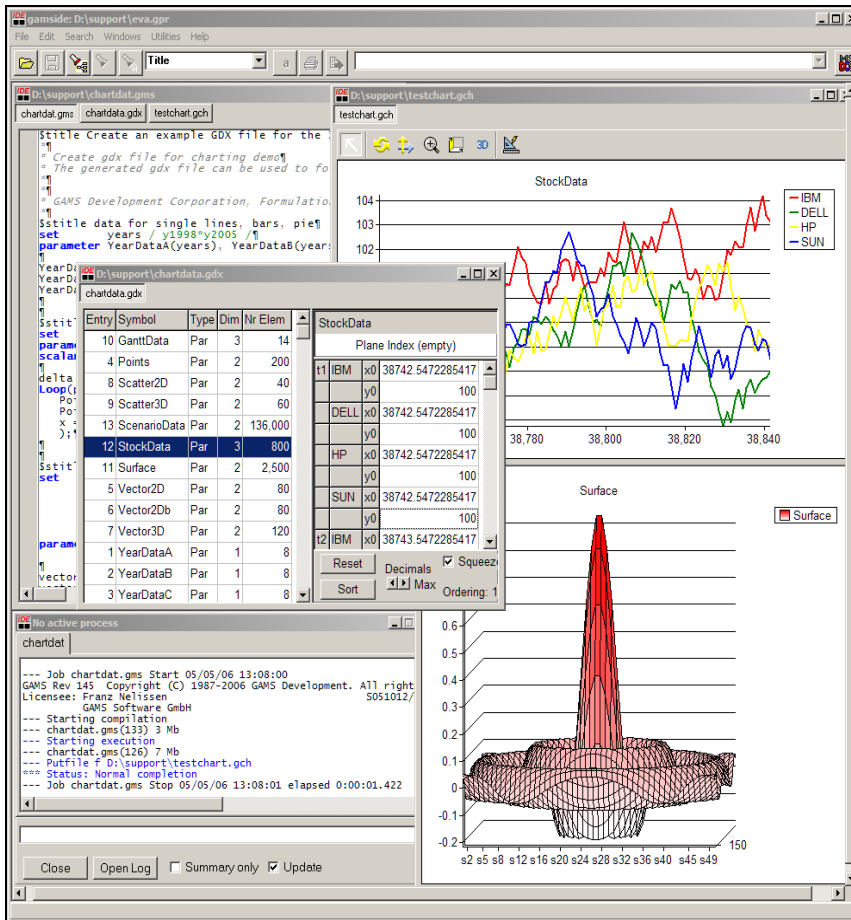
- Fast exchange of data
- Syntactical check on data before model starts
- Data Exchange at any stage (Compile and Run-time)
- Platform Independent
- Direct GDX interfaces and general API
- Scenario Management Support
- Full Support of Batch Runs

GDX Tools





GAMS at a Glance



The GAMS/BASE Module

Free Solvers

- Convert
- EMP/JAMS, LOGMIP, NLPEC
- BENCH, EXAMINER, GAMSCHK
- BDMLP, LS, and MILES
- COIN-OR
Cbc, IpOpt, BonMin, Couenne
- Glpk, Scip (academic only)



New GAMS Distribution 23.5.2

Released August, 18th

www.gams.com/download

Download GAMS Distribution 23.5.2 - August 18, 2010

Note: To deliver GAMS with the best performance we are using the [Amazon CloudFront](#) web service, a global network of edge locations for content delivery.

Please consult the [release notes](#) before downloading a system. The installation notes for [Windows](#) and [UNIX](#) and the complete [system documentation](#) are included in any system.

Windows

[Windows 32 bit](#) Windows 7, Windows Vista, Windows XP, Windows Server 2008, Windows Server 2003, and compatible on AMD- or Intel-based (x86_32) architectures

[Windows 64 bit](#) Windows 7 x64, Windows Vista x64, Windows XP x64, Windows Server 2008 x64, Windows Server 2003 x64, and compatible on AMD- or Intel-based (x64_64) architectures.

Unix

[AIX](#) AIX 5.3 or higher, PowerPC chip, 64 bit (ppc_64)

[Digital UNIX](#) Digital Unix 4 or higher on DEC Alpha (axp_64). Please note that the current GAMS distribution for Digital Unix is 22.7

[HP-UX](#) HP-UX 11 or higher on HP PA-RISC (hpux_32). Please note that the current GAMS distribution for HP-UX is 22.1

[Linux 32 bit](#) AMD- or Intel-based (x86_32) Intel-based 32-bit Linux systems. Most likely, these will have a 2.4.X kernel or higher. The software was built with Intel's Linux compilers, ver 11.1 or higher.

[Linux 64 bit](#) AMD- or Intel-based (x64_64) Linux systems. These were built on a 2.6 kernel with Intel's Linux compilers, ver 11.1 or higher.

[IRIX](#) IRIX 6.2 or higher on SGI MIPS (sgi_32). Please note that the current GAMS distribution for IRIX is 22.3

[Mac OS X Intel 32 bit](#) Macintosh Intel-based systems (x86_32) built on Darwin 9. Please note that this is a Mac OS X Terminal application and must be installed and executed using the command line interface. → [Additional Information](#)

[Mac OS X Intel 64 bit](#) Macintosh Intel-based systems (x64_64) built on Darwin 9. Please note that this is a Mac OS X Terminal application and must be installed and executed using the command line interface. → [Additional Information](#)

[Mac OS X PPC](#) Macintosh PowerPC-based systems for G4/G5 processors (ppc_fat) built on Darwin 8.4. Please note that the current GAMS distribution for Mac OS X PPC is 23.3.3 and that this is a Mac OS X Terminal application and must be installed and executed using the command line interface. [Additional Information](#)

[Solaris SPARC 32 bit](#) Solaris 2.8 or higher on SUN Sparc (sparc_32). Missing [Fortran Run-Time Environment](#)?

[Solaris SPARC 64 bit](#) Solaris 2.8 or higher on SUN Sparc (sparc_64)

[Solaris x64 64 bit](#) Solaris 10 or higher on AMD- or Intel-based 64-bit (x64_64)

Please also visit the information about the [distribution history, changes, and incremental updates](#). For older distributions please follow [this link](#). There are some [mailing lists](#), which will inform you about forthcoming releases, provide additional information, and are useful for questions about GAMS and modeling issues.



New GAMS Distribution 23.5 cont'd

– Solver updates

- BARON 9.0.6
- CPLEX 12.2
- GUROBI 3.0.1
- LINDOGLOBAL 6.1.1
- MOSEK 6
- SCIP 1.2
- XPRESS 20.00
- Coin-OR

(CBC 2.4, Bonmin 1.3, Couenne 0.3, Ipopt 3.8, Glpk 4.43, OS 2.1)

– New platforms

- 64bit Intel Mac
- 64bit AIX

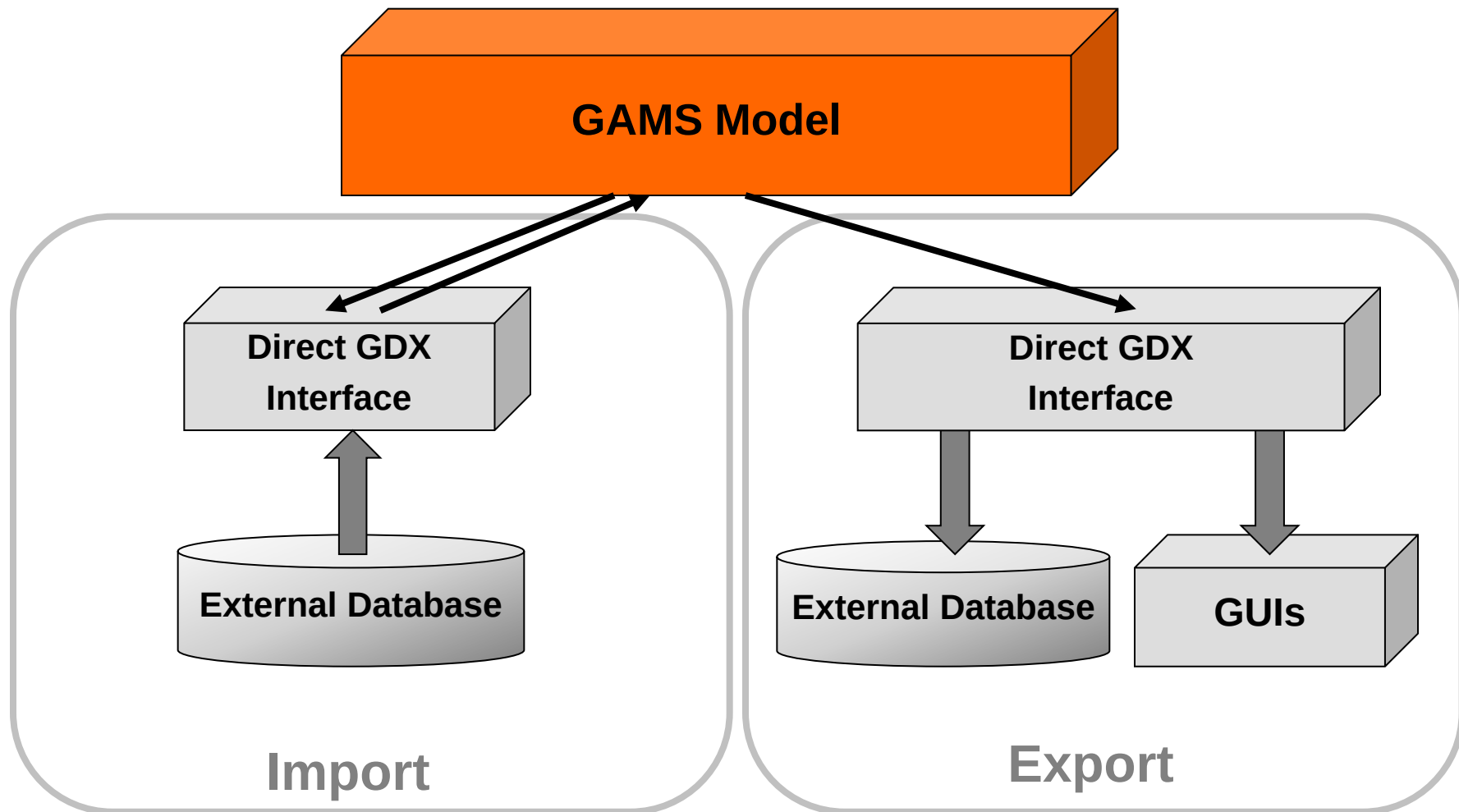


New GAMS Distribution 23.5 cont'd

- Free Coin-OR OSI-based links to **CPLEX, GUROBI, MOSEK and XPRESS**
- **GAMS on Amazon EC2**
- **GAMS API's**
 - Improvements on all frontiers
 - Now supported: Python
 - Examples/Documentation
 - Project and configuration files
- ...

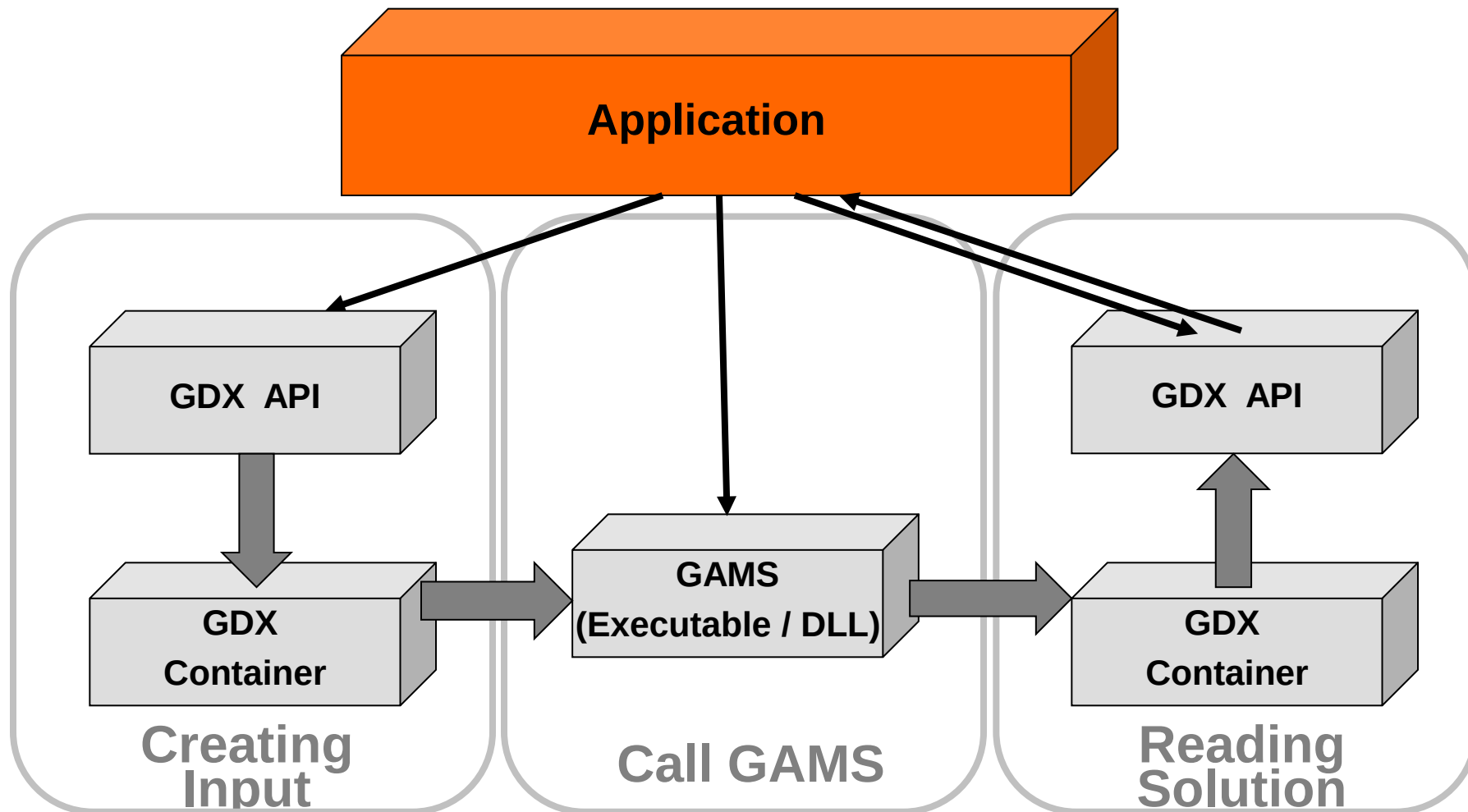


Interfacing: GAMS in Control





Interfacing: Application in Control





GAMS Application Programming Interfaces

GAMS in Control

VS.

Application in Control



- Need to support a variety of applications
 - Web application (server side)
 - Application Builder
 - Oracle, Eclipse, .NET, ...
 - Regular Programming language C(++), C#, Java, VB, Fortran, Python, ...
 - MS Office Application / VBA



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



Calling GAMS from 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)

```

```

from gdxcc import *
from gamsxcc import *
from optcc import *
import sys
import os

```

Creating Input for GAMS Model

Callout to GAMS

Reading Solution from GAMS Model



Calling GAMS from Python cont'd

Creating Input for GAMS Model

```
def writeModelData(gdxHandle, fngdxfile):
    ret, errNr = gdxOpenWrite(gdxHandle, fngdxfile, "Example1")
    if errNr:
        print "*** Error gdxOpenWrite: " + gdxErrorStr(gdxHandle, errNr)[1]
        return False

    if not gdxDataWriteStrStart(gdxHandle, "Demand", "Demand data", 1, GMS_DT_PAR, 0):
        reportGDSError(gdxHandle, "DataWriteStrStart")

    values = doubleArray(GMS_VAL_MAX)
    values[GMS_VAL_LEVEL] = 324.0
    gdxDataWriteStr(gdxHandle, ["New-York"], values)
    values[GMS_VAL_LEVEL] = 299.0
    gdxDataWriteStr(gdxHandle, ["Chicago"], values)
    values[GMS_VAL_LEVEL] = 274.0
    gdxDataWriteStr(gdxHandle, ["Topeka"], values)

    if not gdxDataWriteDone(gdxHandle):
        reportGDSError(gdxHandle, "WriteData")

    errNr = gdxGetLastError(gdxHandle)
    if errNr:
        print "*** Error while writing GDX File: " + gdxErrorStr(gdxHandle, errNr)[1]
        return False

    errNr = gdxClose(gdxHandle)
    if errNr:
        print "*** Error gdxClose: " + gdxErrorStr(gdxHandle, errNr)[1]
        return False

    return True
```



Calling GAMS from Python cont'd

Callout to GAMS

```
- def callGams(gamsxHandle, optHandle, sysDir):  
...     deffile = sysDir + "\\optgams.def"  
...  
...     if optReadDefinition(optHandle, deffile):  
...         print "*** Error ReadDefinition, cannot read def file:" + deffile  
...         return False  
...  
...     optSetStrStr(optHandle, "SysDir", sysDir)  
...     optSetStrStr(optHandle, "Input", "model2.gms")  
...     optSetIntStr(optHandle, "LogOption", 3)  
...     ret = gamsxRunExecDLL(gamsxHandle, optHandleToPtr(optHandle), sysDir, 1)  
-     if ret[0] != 0:  
...         print "*** Error RunExecDLL: Error in GAMS call = " + str(ret[1])  
...         return False  
...  
...     return True
```



Calling GAMS from Python cont'd

Reading Solution from GAMS Model

```
def readSolutionData(gdxHandle, fngdxfile):  
    errNr = gdxOpenRead(gdxHandle, fngdxfile)[1]  
    if errNr:  
        print "**** Error OpenRead: " + gdxErrorStr(gdxHandle, errNr)[1]  
        return False  
  
    ret, varNr = gdxFindSymbol(gdxHandle, "result")  
    if not ret:  
        print "*** Error FindSymbol: Could not find variable result"  
        return False  
  
    ret, symName, dim, varType = gdxSymbolInfo(gdxHandle, varNr)  
    if dim != 2 or varType != GMS_DT_VAR:  
        print "**** Error SymbolInfo: result is not a two dimensional variable"  
        return False  
  
    ret, nrRecs = gdxDataReadStrStart(gdxHandle, varNr)  
    if not ret:  
        reportGDSError(gdxHandle, "DataReadStrStart")  
        return False  
  
    ...
```




Calling GAMS from Excel (VBA)

chp2.xls [Compatibility Mode] - Microsoft Excel

Home Insert Page Layout Formulas Data Review View

Paste Cut Copy Format Painter Clipboard

Calibri 11 Font

Wrap Text Alignment

General Number

Conditional Formatting Styles

Format as Table Styles

Cell Styles

Insert Delete Format Cells

AutoSum Fill Clear Sort & Find & Filter Select Editing

AF40

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1		SG	SG	HTB	GT	GT	HRB	HE	HB	ST	HC												
2		0	1	1	0	1	1	1	1	1	1												
3	Gas			-39.6	-54.69	-85.67																	
4	Coal	-17.62	-35.24																				
5	WasteHeat				36.43	53.2	-42.56	-10.64	-53.2														
6	Steam	5.5	11				10.72			-16.7													
7	Exhaust									16.7	-16.7												
8	Heat			18				8.6184			39.84												
9	Electricity	-0.25	-0.5		13.77	29.55				10.39													
10																							
11																							
12																							
13																							
14																							
15																							
16																							
17																							
18																							
19	GAMS Directory:	c:\program files\GAMS23.5\																					
20	Working Directory:	c:\tmp\																					
21	Solver:	CPLEX																					
22																							
23																							
24																							
25																							
26																							
27																							

Solution - Generated Heat & Electricity

Heat Electricity

Solver: CPLEX
 Equations: 14786 Variables: 19489
 Model Status: 8 Integer Solution
 Solver Status: 1 Normal Completion
 Iterations: 8704 Solve Time: 0.50

 Objective Value: 834036
 No. of Hours GT is ON: 530
 No. of GT Starts: 8

SOLVE **Clear Solution**



The GAMS Macro Facility

- Basic Definition

- `$macro name` `macro body`
- `$macro name(arg1,...)` `macro body with tokens arg1,...`

- Multi-Argument Example

```
$macro ratio(a,b) a/b
```

```
z = ratio(x1,x2);
```

➔ `z = x1/x2;`

- Macros within Macros

```
$macro product(a,b) a*b
```

```
$macro addup(i,x,z) sum(i,product(x(i),z))
```

```
z = addup(j,a1,x1);
```

➔ `z = sum(j,a1(j)*x1);`



The GAMS Macro Facility (contd.)

- Careful expansion (&)

```
$macro f(i)      sum(j, x(i,j))
$macro equ(q)    equation equ_&q; equ_&q.. q =e= 0;
equ(f(i))
```

➔ `equation equ_f(i); equ_f(i).. sum(j, x(i,j)) =e= 0;`

- Removing outer set of quotes (&&)

```
$macro d(q)      display &&q;
d('"here it is", i, k')
```

➔ `display "here it is", i, k;`

```
$macro dd(q)     &&q)
z=dd('sum(j, a1(j)');
```

➔ `z=sum(j, a1(j));`



Enhanced Data Statements

- Allow initial values for equations and variables
- Follow the syntax for list and table data statement for parameters by adding an additional dimension to specify the specific data attribute

Variable table $x(i,j)$ initial values

		l	m
seattle.	new-york	50	
seattle.	Chicago	300	
seattle.	topeka		0.36
san-diego.	new-york	275	
san-diego.	Topeka	275	
san-diego.	chicago		0.009

;



Matrix Utilities

- INVERT
 - Calculates the inverse of a matrix
- CHOLSKY
 - Computes the Cholesky factors of a symmetric positive-definite matrix
- EIGENVALUE
 - Computes the eigenvalues of a symmetric matrix
- EIGENVECTOR
 - Computes the eigenvalues and eigenvectors of a symmetric matrix



How to stay Up To Date

<http://www.gams.com/maillist/>

[\[Home](#) | [Support](#) | [Sales](#) | [Solvers](#) | [Documentation](#) | [Model Libraries](#) | [Search](#) | [Contact Us \]](#)

The GAMS Mailing List

GAMS users worldwide use the list name GAMS-L to exchange information about GAMS. GAMS-L is open to everyone around the world and can easily be reached via the Internet.

[Subscribe \(and more information\)](#)

Bruce McCarl's GAMS Newsletter

With his newsletter Bruce McCarl wants to provide some additional information on the use and features which emerge as GAMS develops. He intends to periodically issue a very short newsletter that informs people of things that are new and or under documented as well as opportunities to learn more about GAMS features and usage.

[Archive/Subscribe/Unsubscribe](#)

The GAMS Release Mailing List

For people interested in receiving the latest information about new GAMS releases and trying out beta releases.

[Subscribe/Unsubscribe](#)

Please visit us at our booth in the exhibit area!



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

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