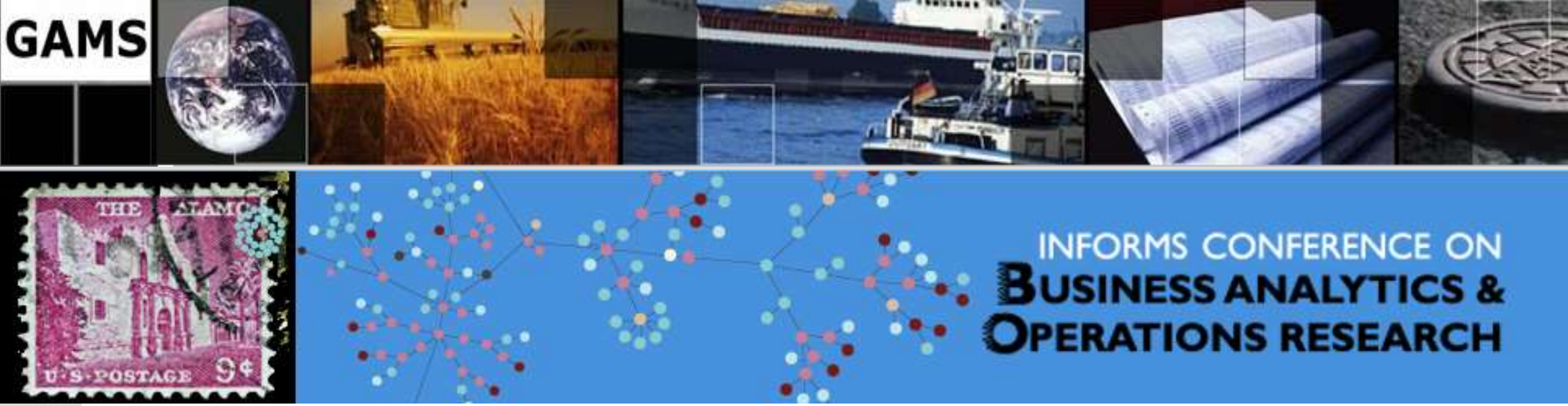


Deploying Optimization Applications
- Concepts and Challenges -

Franz Nelissen FNelissen@gams.com

GAMS Development Corp.

GAMS Software GmbH

www.gams.com

San Antonio, TX, April 2013

Company Background

- Roots: World Bank, 1976
- Went commercial in 1987
- GAMS Development Corporation (Washington, Houston)
- GAMS Software GmbH (Frechen, Braunschweig)
- Tool Provider: **G**eneral **A**lgebraic **M**odeling **S**ystem

Agenda

- What is GAMS?
- Model Deployment
- Recent Enhancements

GAMS

What is GAMS?

GAMS at a Glance

Design Principles

Algebraic Modeling Languages (AML)

What's that?

http://en.wikipedia.org/wiki/Algebraic_modeling_language

- High-level **computer programming languages** for the formulation of **complex mathematical optimization problems**
- **Notation similar to algebraic notation**: Concise and readable definition of problems in the domain of optimization
- **Do not solve problems directly**, but ready-for-use links to state-of-the-art algorithms

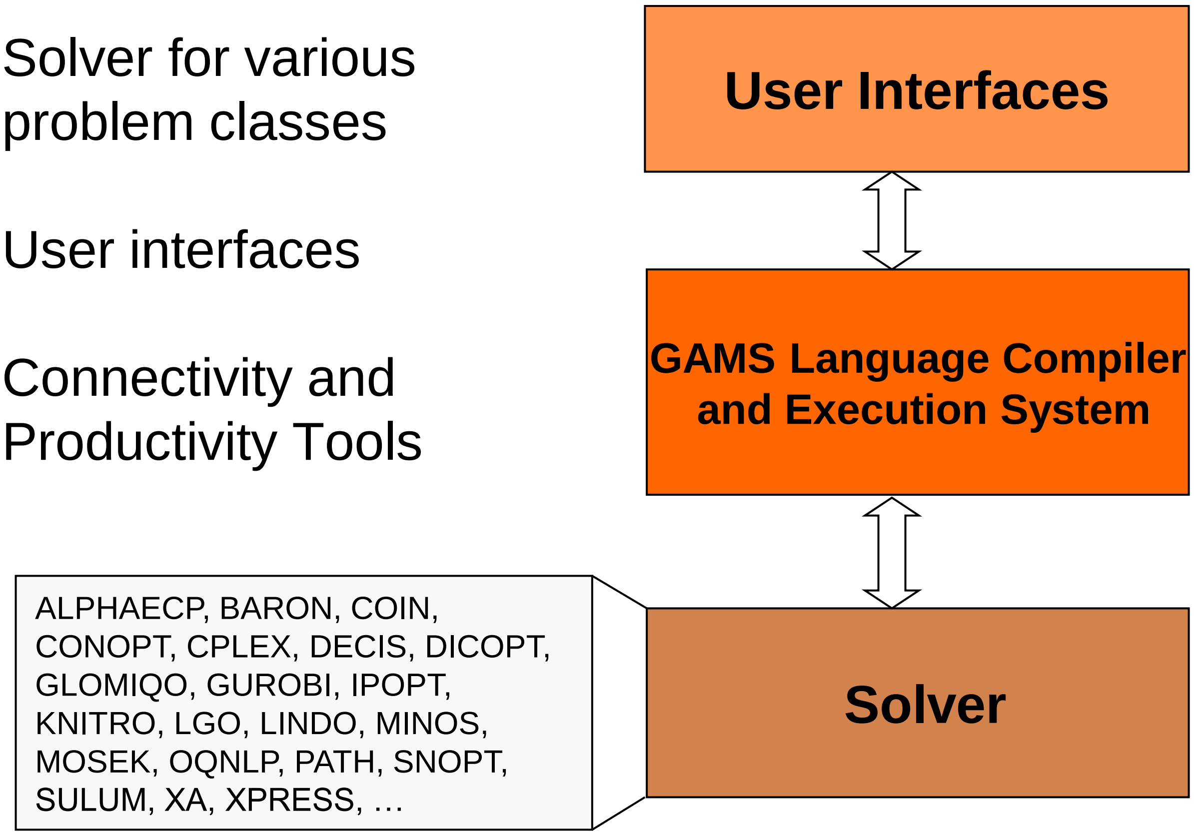
Impact of Algebraic Modeling Languages

- Simplified model development, changes, and transfer
- Rapid prototyping
- Added value to existing applications
- Increased productivity, quality, reliability and maintainability

→ Important vehicle to make mathematical optimization available to a broader audience, e.g. domain specific experts

GAMS: Core Elements

- Language compiler and execution system
- Solver for various problem classes
- User interfaces
- Connectivity and Productivity Tools



Connectivity Tools

- Uniform Data Exchange (ASCII, binary)
- Low-level and object-oriented API
- Ext. programs: EXCEL, MATLAB, R, …

Productivity Tools

- **I**ntegrated **D**evelopment **E**nvironment (**IDE**)
- Integrated Data Browser and Charting Engine
- Model Debugger and Profiler
- Model Libraries
- Benchmarking and Deployment
- Quality Assurance and Testing System
- Data and Model Encryption
- Grid Computing
- Scenario Reduction

GAMS: Broad Range of Application Areas*

- Agricultural Economics
- Chemical Engineering
- Econometrics
- Environmental Economics
- Finance
- International Trade
- Macro Economics
- Management Science/OR
- Micro Economics
- Applied General Equilibrium
- Economic Development
- Energy
- Engineering
- Forestry
- Logistics
- Military
- Mathematics
- Physics

*** Illustrative examples in the GAMS Model Library**

GAMS: Broad User Network

To get your own ClustrMap, register here

distance in which individuals are clustered
Dot sizes: = 1,000+ = 100 - 999 = 10 - 99 = 1 - 9 visits

= Recent [?]

18 Mar 2013 to 24 Mar 2013: 5,071 visits shown above

Statistics updated 24 Mar 2013@08:38GMT: 5,094 visits [?]
Total since 9 Sep 2010: 569,609. Previous 24hrs: 773.

Notes | Country totals =>

Current Country Totals
From 18 Mar 2013 to 24 Mar 2013

United States (US)	1,572
India (IN)	998
Finland (FI)	278
Turkey (TR)	152
Pakistan (PK)	123
Bangladesh (BD)	109
Spain (ES)	93

GAMS: Monthly System Downloads

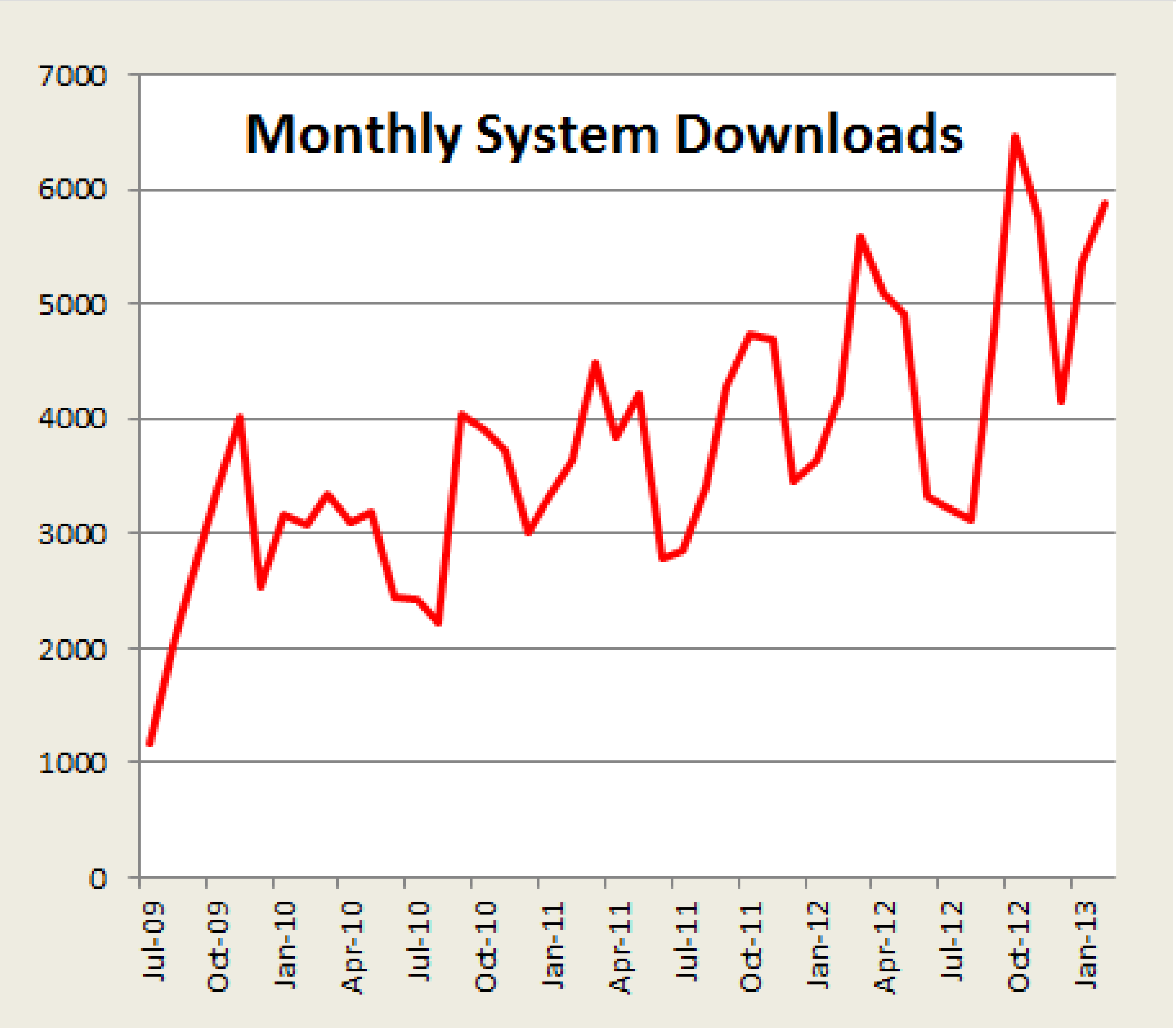
http://www.gams.com/download/

GAMS

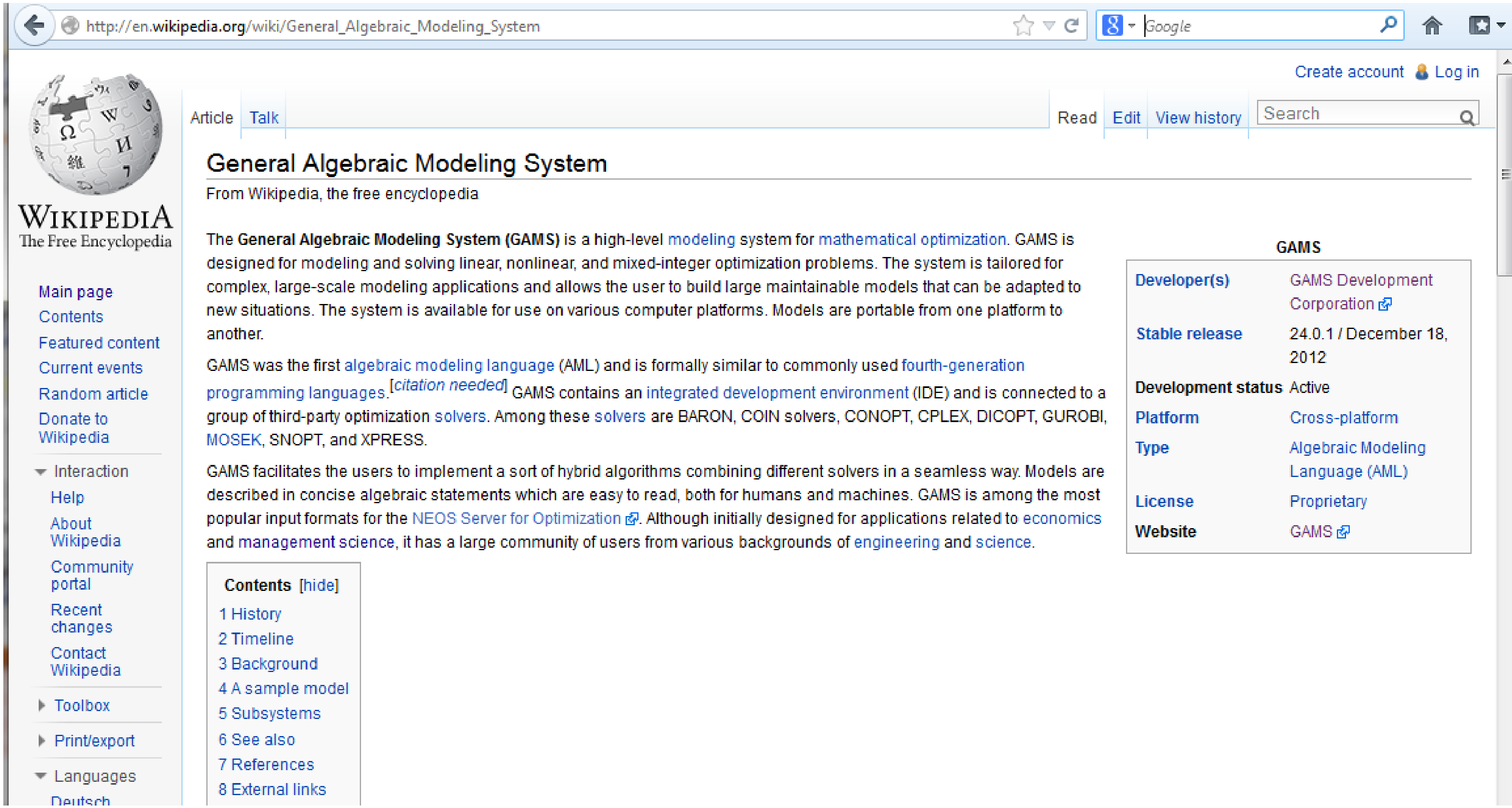
Download GAMS Distribution 24.0.2

Please consult the release notes before downloading a system. The installation notes f

Windows	
Windows 32 bit	Windows 8, Windows 7, Windows Vista, Windows XP, Win
Windows 64 bit	Windows 8 x64, Windows 7 x64, Windows Vista x64, Wind
Unix	
AIX	AIX 5.3 or higher, PowerPC chip, 64 bit (ppc_64)
Linux 32 bit	AMD- or Intel-based 32-bit Linux systems. The software wa
Linux 64 bit	AMD- or Intel-based 64-bit Linux systems (x86_64). The sc
Mac OS X Intel 64 bit	Macintosh Intel-based systems (x64_64) built on Darwin 10
Solaris SPARC 32 bit	Solaris 2.8 or higher on SUN Sparc (sparc_32). Missing For
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
Wine	
Linux/Wine (beta)	AMD- or Intel-based Linux systems. The software uses the
Mac/Wine (beta)	For more information please visit this page.



GAMS: Wikipedia



http://en.wikipedia.org/wiki/General_Algebraic_Modeling_System

Create account Log in

Article Talk Read Edit View history Search

WIKIPEDIA
The Free Encyclopedia

Main page
Contents
Featured content
Current events
Random article
Donate to Wikipedia

Interaction
Help
About Wikipedia
Community portal
Recent changes
Contact Wikipedia

Toolbox
Print/export
Languages
Deutsch

General Algebraic Modeling System

From Wikipedia, the free encyclopedia

The **General Algebraic Modeling System (GAMS)** is a high-level modeling system for mathematical optimization. GAMS is designed for modeling and solving linear, nonlinear, and mixed-integer optimization problems. The system is tailored for complex, large-scale modeling applications and allows the user to build large maintainable models that can be adapted to new situations. The system is available for use on various computer platforms. Models are portable from one platform to another.

GAMS was the first algebraic modeling language (AML) and is formally similar to commonly used fourth-generation programming languages.[citation needed] GAMS contains an integrated development environment (IDE) and is connected to a group of third-party optimization solvers. Among these solvers are BARON, COIN solvers, CONOPT, CPLEX, DICOPT, GUROBI, MOSEK, SNOPT, and XPRESS.

GAMS facilitates the users to implement a sort of hybrid algorithms combining different solvers in a seamless way. Models are described in concise algebraic statements which are easy to read, both for humans and machines. GAMS is among the most popular input formats for the NEOS Server for Optimization. Although initially designed for applications related to economics and management science, it has a large community of users from various backgrounds of engineering and science.

GAMS

Developer(s)	GAMS Development Corporation
Stable release	24.0.1 / December 18, 2012
Development status	Active
Platform	Cross-platform
Type	Algebraic Modeling Language (AML)
License	Proprietary
Website	GAMS

Contents [hide]

What is GAMS?

GAMS at a Glance

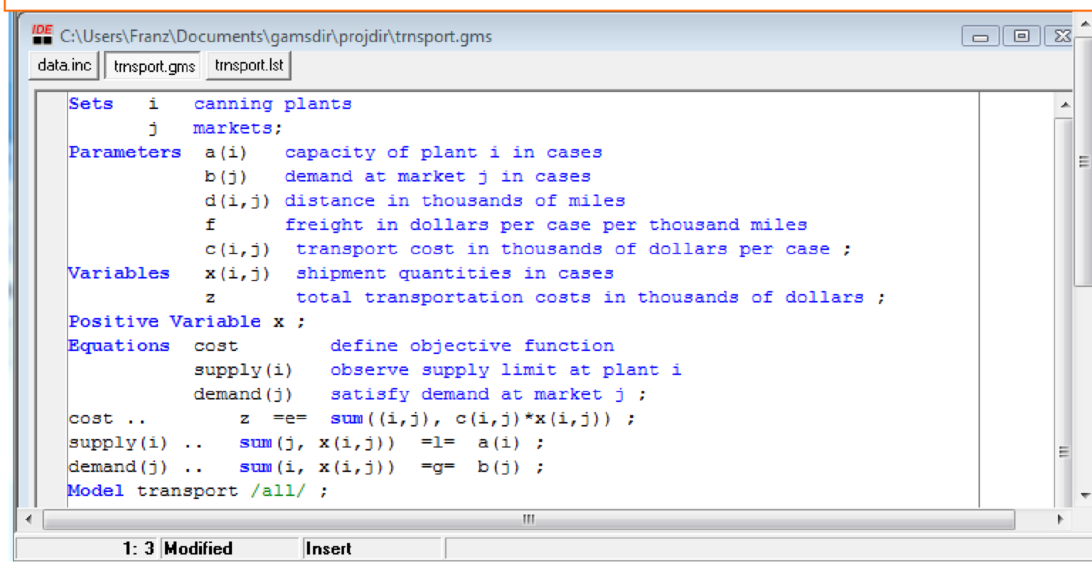
Design Principles

GAMS Design Principles

Universal language: Algebra, similar to mathematical notation

- Algebra (Expressions): model equations
- Relational Algebra (SQL): data manipulation
- Model is communication device and executable documentation of the optimization problem

$$
\begin{array}{llll}
\text{Objective} & \sum_i \sum_j c_{i,j} \times x_{i,j} & \longrightarrow \min & \\
\text{Observe supply limit at plant } i\text{:} & \sum_j x_{i,j} & \leq a_i & \forall i \\
\text{Satisfy demand at market } j\text{:} & \sum_i x_{i,j} & \geq b_j & \forall j \\
 & x_{i,j} & \geq 0 & \forall i,j
\end{array}
$$




```
Sets     i    canning plants
         j    markets;
Parameters  a(i)    capacity of plant i in cases
            b(j)    demand at market j in cases
            d(i,j)  distance in thousands of miles
            f       freight in dollars per case per thousand miles
            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) ..    sum(j, x(i,j))   =l=  a(i) ;
demand(j) ..    sum(i, x(i,j))   =g=  b(j) ;
Model transport /all/ ;
```

GAMS Design Principles

Balanced mix of declarative and procedural elements:

- Declarative: sets, parameters, variables, equations, models
- but also procedural elements like:
  - For

```
scalar scen;
for (scen=1 t0 10 by 0.5,
     f = 10*scen; c(i,j)= f * d(i,j) /1000;
     solve transport using lp minimizing z;
     display z.l;
    );
```

  - Loop/If

```
loop (h,
     if (work(h),
          pay(i,h) = 0.6 * pay(i,h);
     else
          pay(i,h) = 1.5 * pay(i,h);
        );
      );
```

  - Macros

```
$macro eqcon(y)  sqr{1-sqr[x('1')]*sqr[y]} - x('1')*sqr[y] - sqr[x('2')] + x('2')
eqconlow..         eqcon(y('1')) =e= vio;
eqconlbd(dlbd)..   eqcon(ylbd('1',dlbd)) =l= 0;
eqconubd(dubd)..   eqcon(yubd('1',dubd)) =l= -epsilon;
```

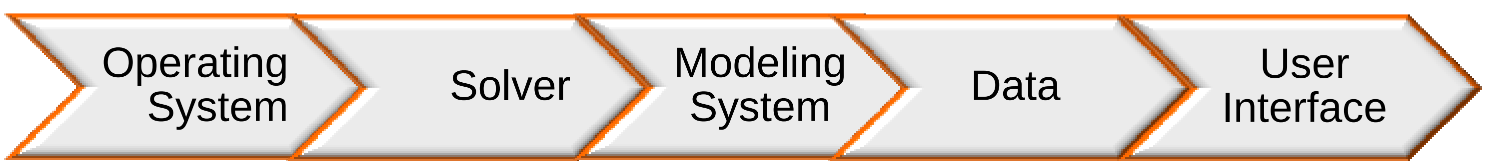
  - User defined functions

```
$funclibin fitlib
function fitp /fitlib.fitparam/
         fit  /fitlib.fitfunc /;
```

GAMS Design Principles

Independent Layers:

- Model and operating system
- Model and solution methods (solver)
- Model and modeling system
- Model and data
- Model and user interface



Independence of Model and Operating System

Supported Platforms

Solver/Platform availability - 24.0 December 24, 2012									
	x86 MS Windows	x86_64 MS Windows	x86 Linux	x86_64 Linux	Sun Sparc SOLARIS	Sun Sparc64 SOLARIS	Sun Intel SOLARIS	IBM RS-6000 AIX 5.3	Mac x86_64 Darwin
ALPHAECP	✔	✔	✔	✔	✔	✔	✔	✔	✔
BARON 11.7	✔	✔	✔	✔					
BDMLP	✔	✔	✔	✔	✔	✔	✔	✔	✔
COIN-OR	✔	✔	✔	✔			✔		✔
CONOPT 3	✔	✔	✔	✔	✔	✔	✔	✔	✔
CPLEX 12.5	✔	✔	✔	✔	✔	✔	✔	✔	✔
DECIS	✔	✔	✔	✔	✔	32bit			
DICOPT	✔	✔	✔	✔	✔	✔	✔	✔	✔
GLOMIQO 2.1	✔	✔	✔	✔					
GUROBI 5.0	✔	✔	✔	✔				✔	✔
KNITRO 8.0	✔	✔	✔	✔			✔		✔
LINDO 7.0	✔	✔	✔	✔			✔		✔
LINDOGLOBAL 7.0	✔	✔	✔	✔	6.0	6.0	✔		✔
LGO	✔	✔	✔	✔	✔	✔	✔		✔
MILES	✔	✔	✔	✔	✔	✔	✔	✔	✔
MINOS	✔	✔	✔	✔	✔	✔	✔	✔	✔
MOSEK 6	✔	✔	✔	✔			✔		✔
MPSGE	✔	✔	✔	✔	✔	✔	✔	✔	✔
MSNLP	✔	✔	✔	✔	✔	32bit			✔
NLPEC	✔	✔	✔	✔	✔	✔	✔	✔	✔
OQNLP	✔	32bit	✔	32bit					
PATH	✔	✔	✔	✔	✔	✔	✔	✔	✔
SBB	✔	✔	✔	✔	✔	✔	✔	✔	✔
SCIP 3.0	✔	✔	✔	✔			✔		✔
SNOPT	✔	✔	✔	✔	✔	✔	✔	✔	✔
SOPLEX 1.7	✔	✔	✔	✔			✔		✔
SULUM 1.0	✔	✔	✔	✔					
[illegible]	✔	✔	✔	✔					
[illegible]	✔	✔	✔	✔	✔	✔	✔	✔	✔
Fo[illegible]r versions of operating systems and solvers. Please call.									

Operating System

Independence of Model and Solution Methods

Supported Model Types and Solvers

Solver/Model type availability - 24.0 December 24, 2012

	LP	MIP	NLP	MCP	MPEC	CNS	DNLP	MINLP	QCP	MIQCP	*Stoch.*	*Global*
ALPHAECP								✔		✔		
BARON 11.7	✔	✔	✔				✔	✔	✔	✔		✔
BDMLP	✔	✔										
COIN-OR	✔	✔	✔				✔	✔	✔	✔		✔
CONOPT 3	✔		✔			✔	✔		✔			
CPLEX 12.5	✔	✔							✔	✔		
DECIS	✔										✔	
DICOPT								✔		✔		
GLOMIQO 2.1									✔	✔		✔
GUROBI 5.0	✔	✔										
KNITRO 8.0	✔		✔									
LINDO 7.0	✔	✔	✔									
LINDOGLOBAL 7.0	✔	✔	✔									
LGO	✔		✔									
MILES				✔								
MINOS	✔		✔									
MOSEK 6	✔	✔	✔									
MPSGE												
MSNLP			✔									
NLPEC				✔	✔							
OQNLP			✔									
PATH				✔								
SBB												
SCIP 3.0		✔	✔									
SNOPT	✔		✔									
SOPLEX 1.7	✔											
SULUM 1.0	✔											
XA	✔	✔										
XPRESS	✔	✔							✔	✔		
Contributed Plug&Play solvers												
Kest					✔	✔	✔	✔	✔	✔	✔	✔

LP	Linear Programming
MIP	Mixed-Integer Programming
NLP	Non-Linear Programming
MCP	Mixed Complementarity Problems
MPEC	Mathematical Programs with Equilibrium Constraints
CNS	Constrained Nonlinear Systems
DNLP	Non-Linear Programming with Discontinuous Derivatives
MINLP	Mixed-Integer Non-Linear Programming
QCP	Quadratically Constrained Programs
MIQCP	Mixed Integer Quadratically Constrained Programs

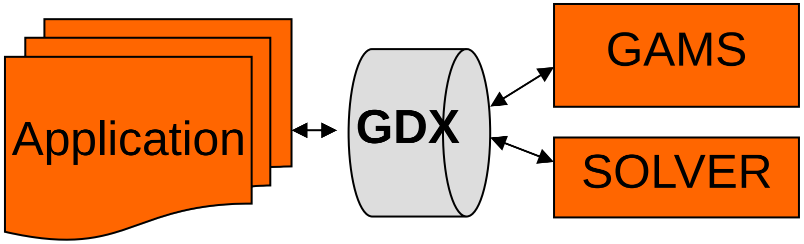
Operating System → **Solver**

Independence of Model and Modeling System

- Don‘t lock users into GAMS System
- Support translation of GAMS code into other environments
- GAMS/CONVERT converts GAMS to AMPL, BARON, CplexLP, MPS, FixedMPS, …
- Component Libraries: Wrap AML components in library/plug-in and make them accessible to .NET/Java/…

Independence of Model and Data

- Declarative Modeling
- Scalability / Sparsity
- Various interfaces (ASCII, GDX)
- GDX serves as data layer between GAMS and other applications



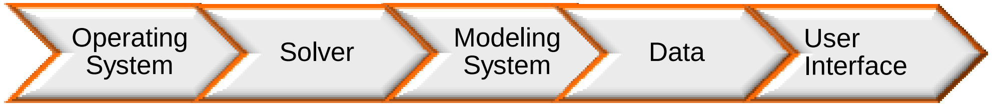
Binary Data Exchange

- Fast exchange of data at any stage
- Binary (no loss of precision)
- Sets, Parameters, Variable- and Equation values (not algebra)
- Can be compressed
- Platform Independent
- Open API for all major languages
- Scenario Management Support
- No license required

GAMS

GAMS Fundamentals: Summary

- Balanced mix of declarative and procedural elements
- Platform independence
- Hassle-free switch of solution methods and solvers
- Independent layers:



→Models benefit from

- Advancing hardware
- New OS / Paradigms (e.g. Cloud)
- Enhanced / new solver technology
- Improved / upcoming interfaces to other systems

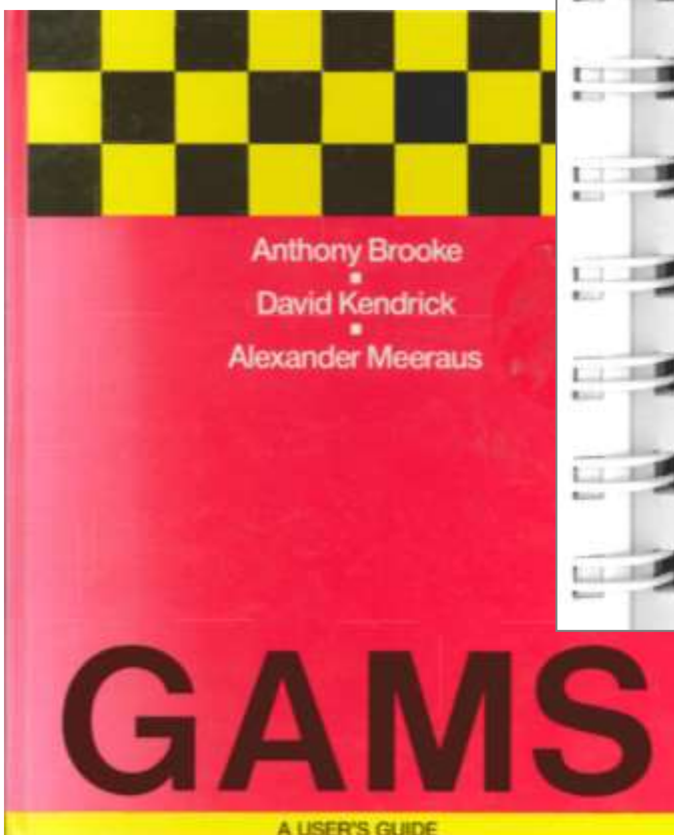
Model Deployment

Change in Focus

Some Examples

Deploying GAMS Models

Change in Focus: Then …



In Table 17.1 we list sizes and attributes of representative models that are "large" in the sense that they are near the limit of what is practical on a personal computer, along with the model generation time (GAMS) and solution time (solver), both in minutes. These examples were run on an 8 MHz AT with an 80287 coprocessor and 640K of RAM. The times shown are to give you a rough idea of what is possible: these are not precisely controlled benchmarks, and we have a host of performance improvements in mind for the near future.

Table 17.1: Problem Characteristics

Name	*Number of Rows*	*Number of Columns*	*Number of Nonzeroes*	*Generation Time*[a]	*Solution Time*[a]	*Iterations*	*Solver*
DINAMICO	318	425	4156	3.0	30.1	628	MINOS
SARF	532	542	3949	37.7	115.8	2775	MINOS
FERTD[b]	458	2968	7252	11.4	28.3	1368	ZOOM
CAMCGE[c]	243	280	1356	0.8	7.0	189	MINOS
GANGES[d]	274	357	1405	1.8	7.3	187	MINOS
YEMCEM[e]	168	258	953	0.9	7.6	600	ZOOM
EGYPT[f]	281	618	3168	4.0	25.3	1551	ZOOM

[a] Measured in minutes.
[b] The problem is too big for MINOS. ZOOM was used instead.
[c] A nonlinear problem. 63% of the non-zeroes are nonlinear.
[d] A nonlinear problem. 58% of the non-zeroes are nonlinear.
[e] A mixed binary problem, with 55 binary variables (solved with a relative termination criterion of 10%).
[f] A linear problem, solved using XMP which is contained within ZOOM.

GAMS Users Guide (1988)

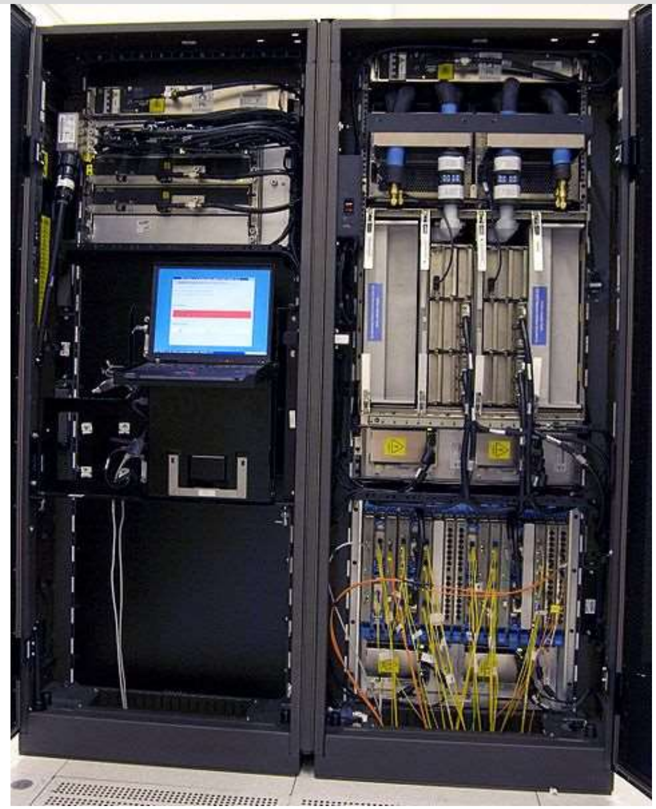
Change in Focus: … and now

	Type	Solution Time (s) 1988	2008	Improvement Factor
camcge	NLP	468	0.031	15097
dinamico	LP	1986	0.125	15888
egypt*	MIP	1758	0.015	117200
fertd*	MIP	2382	0.062	38419
ganges	NLP	546	0.109	5009
sarf	LP	9210	0.139	66259
yemcem*	MIP	510	0.140	3643

* MIP 1988 solver ZOOM, 2008 solver CPLEX

Change in Focus: Past

X,ASGHC2	B,AS,,C2	-1,00000
X,ASGHC2	A,TRA	6,98400
X,ASGHC3	D,,,GH,N	0,33500
X,ASGHC3	R,,,GHC3	1,00000
X,ASGHC3	B,AS,,C3	-1,00000
X,ASGHC3	A,TRA	6,98400
X,ASGHAS	D,,,GH,N	0,20600
X,ASGHAS	R,,,GHAS	1,00000
X,ASGHAS	B,AS,,AS	-1,00000
X,ASGHAS	A,TRA	6,98400
X,ASGHS1	D,,,GH,P	0,15000
X,ASGHS1	R,,,GHS1	1,00000
X,ASGHS1	B,AS,,S1	-1,00000
X,ASGHS1	A,TRA	6,98400
X,ASGHCN	R,,,GHCN	1,00000
X,ASGHCN	B,AS,,CN	-1,00000
X,ASGHCN	A,TRA	6,98400
X,ASKSC1	D,,,KS,N	0,26000
X,ASKSC1	R,,,KSC1	1,00000
X,ASKSC1	B,AS,,C1	-1,00000
X,ASKSC1	A,TRA	7,56000
X,ASKSC2	D,,,KS,N	0,31000
X,ASKSC2	R,,,KSC2	1,00000
X,ASKSC2	B,AS,,C2	-1,00000
X,ASKSC2	A,TRA	7,56000
X,ASKSC3	D,,,KS,N	0,33500
X,ASKSC3	R,,,KSC3	1,00000
X,ASKSC3	B,AS,,C3	-1,00000
X,ASKSC3	A,TRA	7,56000
X,ASKSAS	D,,,KS,N	0,20600
X,ASKSAS	R,,,KSAS	1,00000
X,ASKSAS	B,AS,,AS	-1,00000
X,ASKSAS	A,TRA	7,56000
X,ASKSS1	D,,,KS,P	0,15000
X,ASKSS1	R,,,KSS1	1,00000
X,ASKSS1	B,AS,,S1	-1,00000
X,ASKSS1	A,TRA	7,56000
X,ASKSCN	R,,,KSCN	1,00000
X,ASKSCN	B,AS,,CN	-1,00000



Computation

→Hardware / Algo. limits application

→ Users left out

Change in Focus: Now

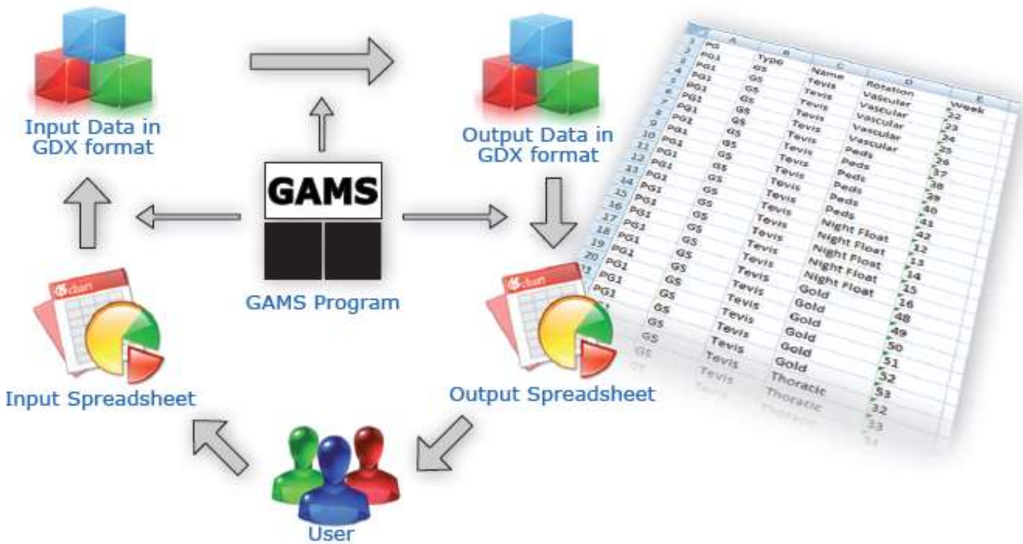
C:\Users\franz\Documents\gamsdir\projdir\epscmmip.gms

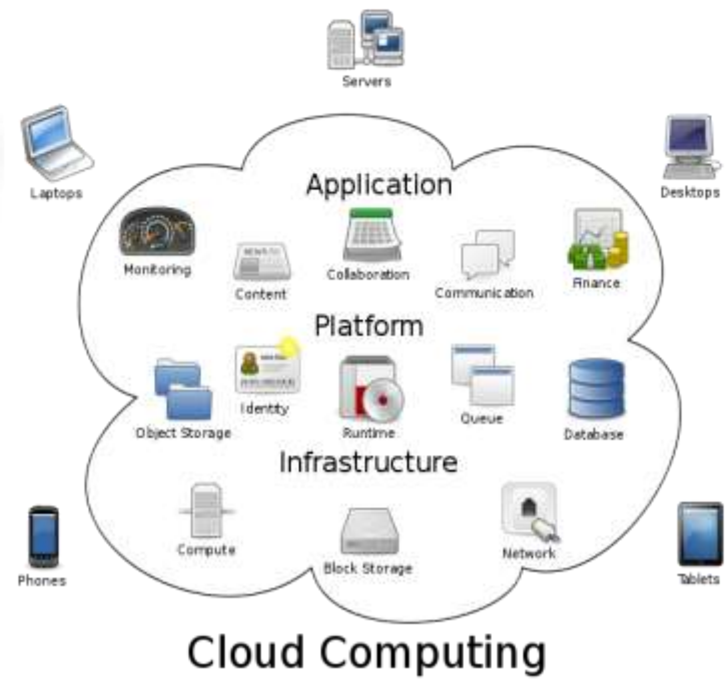
epscmmip.gms

```
posg(km1) = 0; iter=0; infeas=0; start=jnow;

repeat
  rhs(km1) = sum(grid(km1,g)$(numg(g)=posg(km1)), gridrhs(km1,g));
  solve mod_epsmethod maximizing a_objval using mip;
  iter=iter+1;
  if (mod_epsmethod.modelstat<>%ModelStat.Optimal% and
      mod_epsmethod.modelstat<>%ModelStat.Integer Solution%,
    infeas=infeas+1;   // not optimal is in this case infeasible
    put iter:5:0, '  infeasible' /;
    lastZero = 0; loop(km1$(posg(km1)>0 and lastZero=0), lastZero=numk(km1));
    posg(km1)$(numk(km1)<=lastZero) = maxg(km1); // skip all solves for more demanding va
  else
    put iter:5:0;
    loop(k, put z.l(k):12:2);
    jump(km1)=1;
*   find the first off max (obj function that hasn't reach the final grid point).
*   If this obj.fun is k then assign jump for the 1..k-th objective functions
*   The jump is calculated for the innermost objective function (km=1)
    jump(km1)$(numk(km1)=1)=1+floor(sl.L(km1)/step(km1));
    loop(km1$(jump(km1)>1), put '   jump');
    put /;
    );
* Proceed forward in the grid
  firstOffMax = 0;
  loop(km1$(posg(km1)<maxg(km1) and firstOffMax=0),
     posg(km1)=min((posg(km1)+jump(km1)),maxg(km1)); firstOffMax=numk(km1));
  posg(km1)$(numk(km1)<firstOffMax) = 0;
```

Computation	**Model**
→ Users left out	→ Modeling skill limits applications → Users involved

Change in Focus: Now / Future





Computation

→ *Users left out*

Model

→ *Users involved*

Application

→Embedded Models

→ *Users hardly aware of model*

Change in Focus: Fields

Mathematical Optimization ~~*Mathematical Programming*~~ *2010*	***Business Analytics***
• Well defined (narrow) application area	• Broad application area
• Small set of methods (LP, MIP, NLP,…)	• Wide range of methods
• Small amounts of (structured) data	• Huge amounts of data
• Visualization?	• Visualization crucial
• Niche market	• Big market

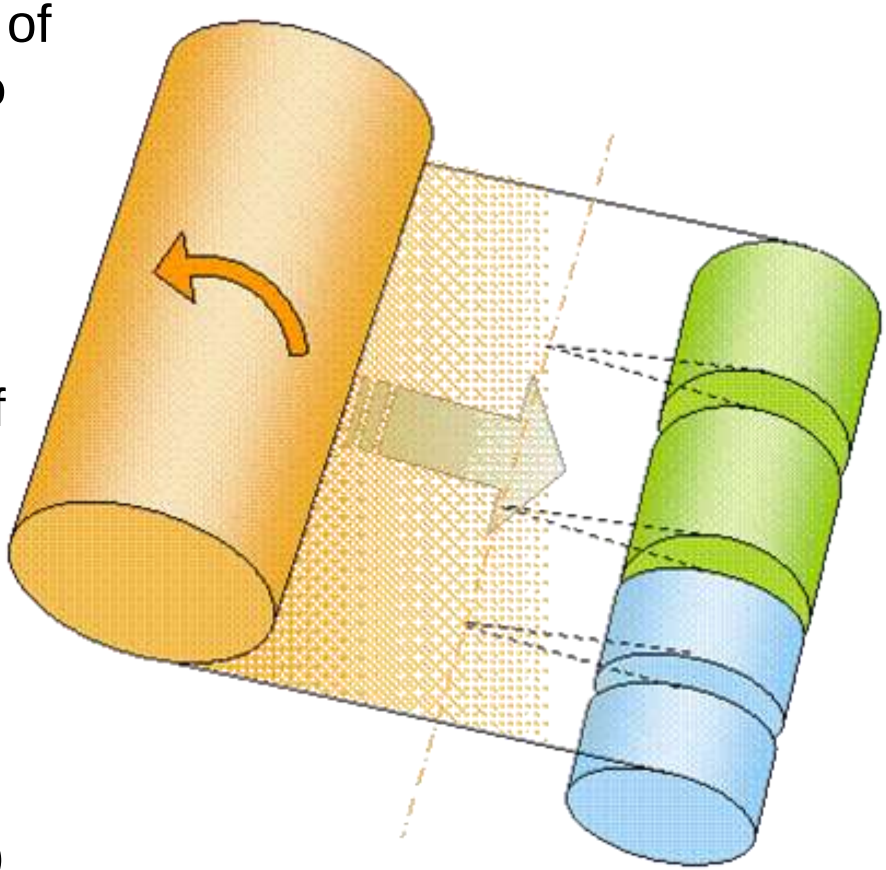
Change in Focus: Looking at Problems

- Cutting Stock Problem: Paper rolls of fixed width (“raw”) must be cut into smaller portions (“finals”).

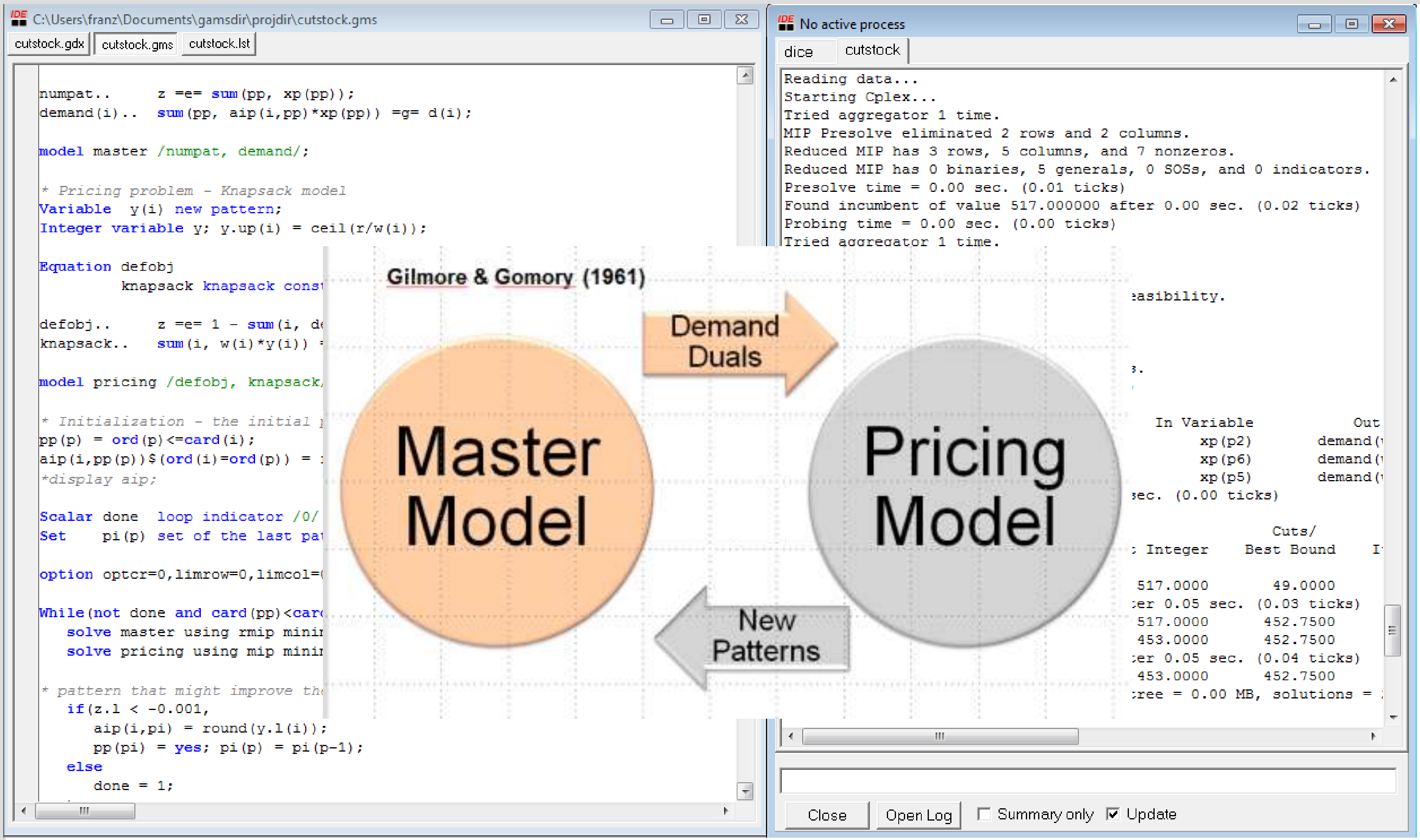
- Input:
  - Width of the raw
  - Demand: Widths and number of finals

- Objective: Minimize the required number of raws

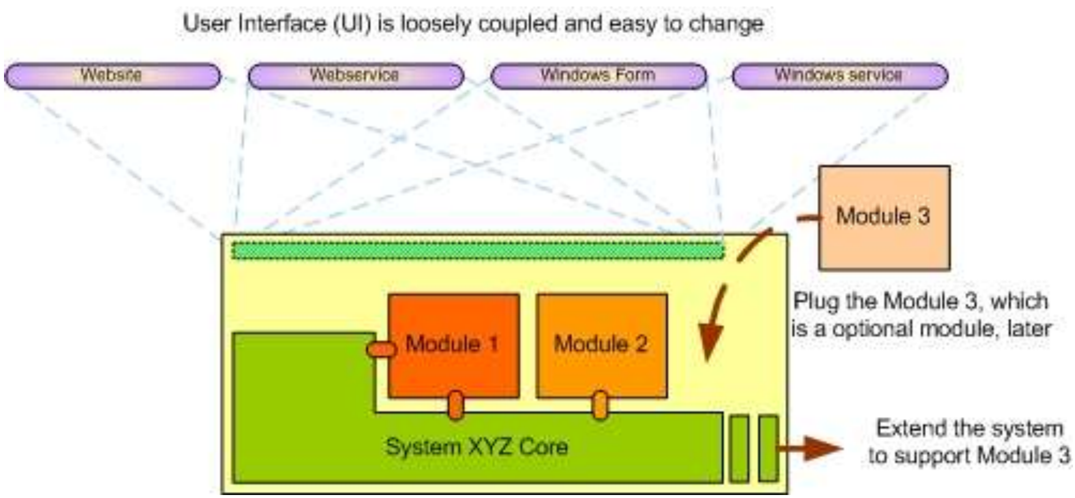
- Output:
  - Combination of cuts (“patterns”)
  - Produced number of each pattern
  - Number of required raws



Change in Focus: Modeler…

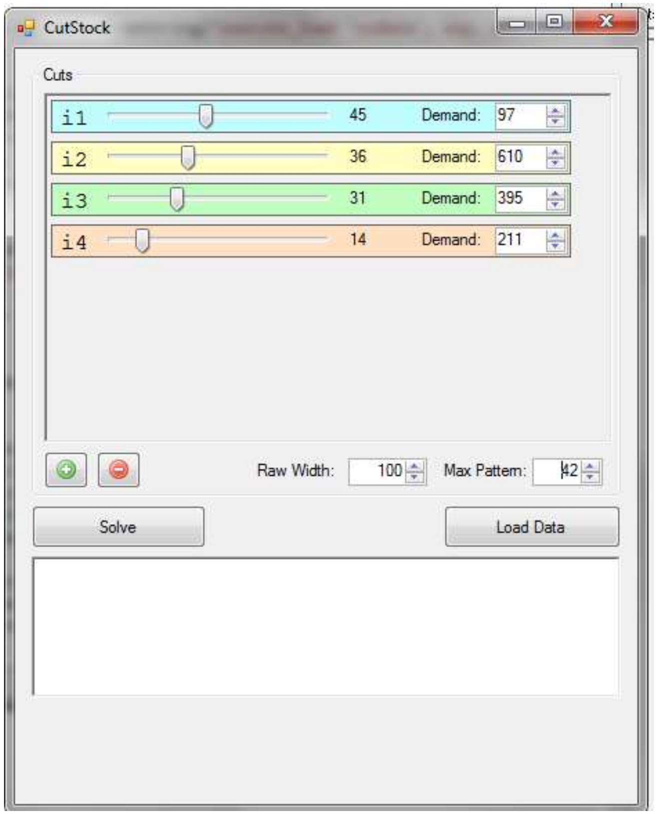


Change in Focus: … Application Developer



- ***Software Architecture***
- ***Object Oriented Design***
- ***Components, Encapsulation, Classes, Data Acess Layer,..***

Rapid Prototyping



Analytic Professionals & IT Professionals

- **Analytic Professionals**
  - INFORMS 10-12k members
  - 64,600 OR analysts in the US (2010, US Bureau of Labor Statistics)
  - INFORMS: First Certified Analytics Professional (CAP) certification, April 7-9, 2013 San Antonio

- **IT Professionals**
  - 3.3 Million IT professionals in the US (2006, Ziff Davis)
  - 15 million IT professionals worldwide (2009, Capers Jones, SPRI)
  - Rapidly changing IT environments (Cloud, RCP, Web, Mobile, …)
  - Certifications for IT professionals are standard

What is special about Optimization?

Optimization

- **May take longer than one is willing to wait**
- **Will eventually fail**

→ Algebraic Modeling Systems provide environment to build robust and fail safe systems

What is special about Optimization?

Optimization models

– **are expensive to develop**
– **may have long lifespan**

→ Modeling Systems & Applications have to be adjusted:

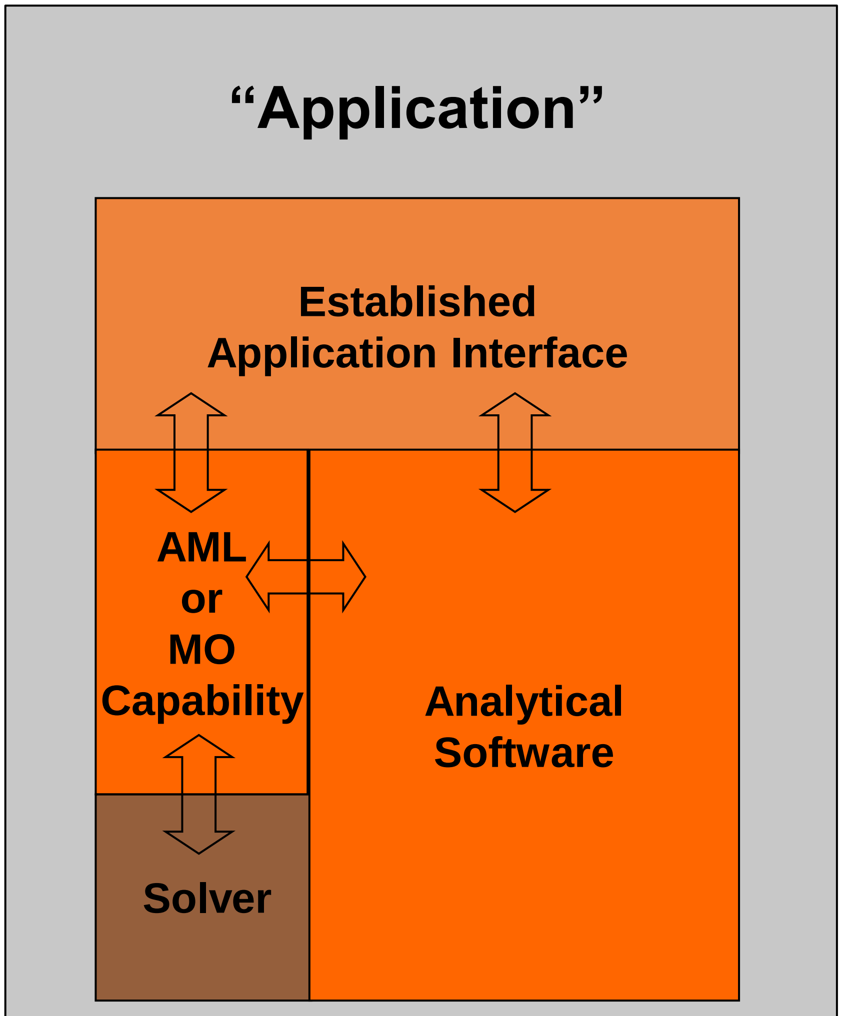
- New computer paradigms (e.g. cloud computing, tablets)
- New solver technology and solution methods
- New graphical user interfaces and deployment environments

→ Minimize risks for (new) clients / management
→ Don’t lock developers and users into a certain environment.
→ Protect user investments
→ Provide cutting edge technology

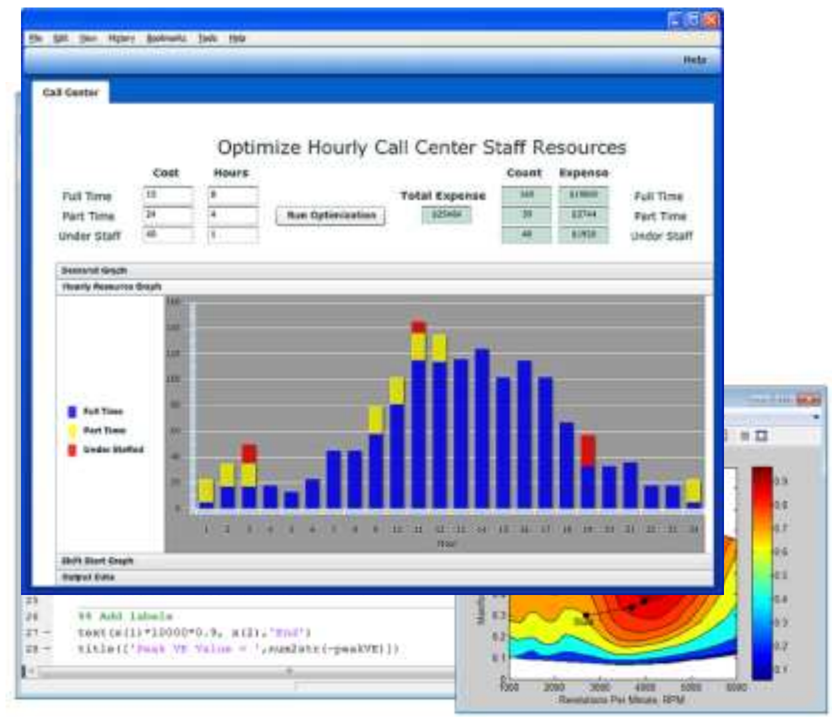
Model Deployment

- Change in Focus
- **Some Examples**
- Deploying GAMS Models

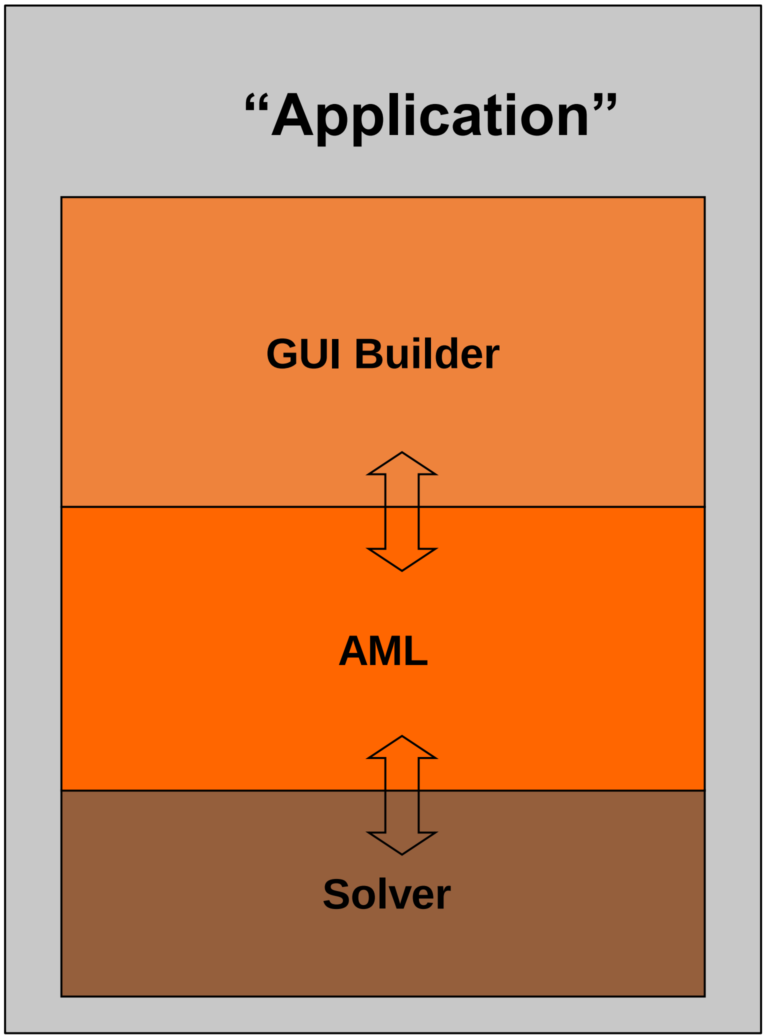
Example – All in One – Top Down



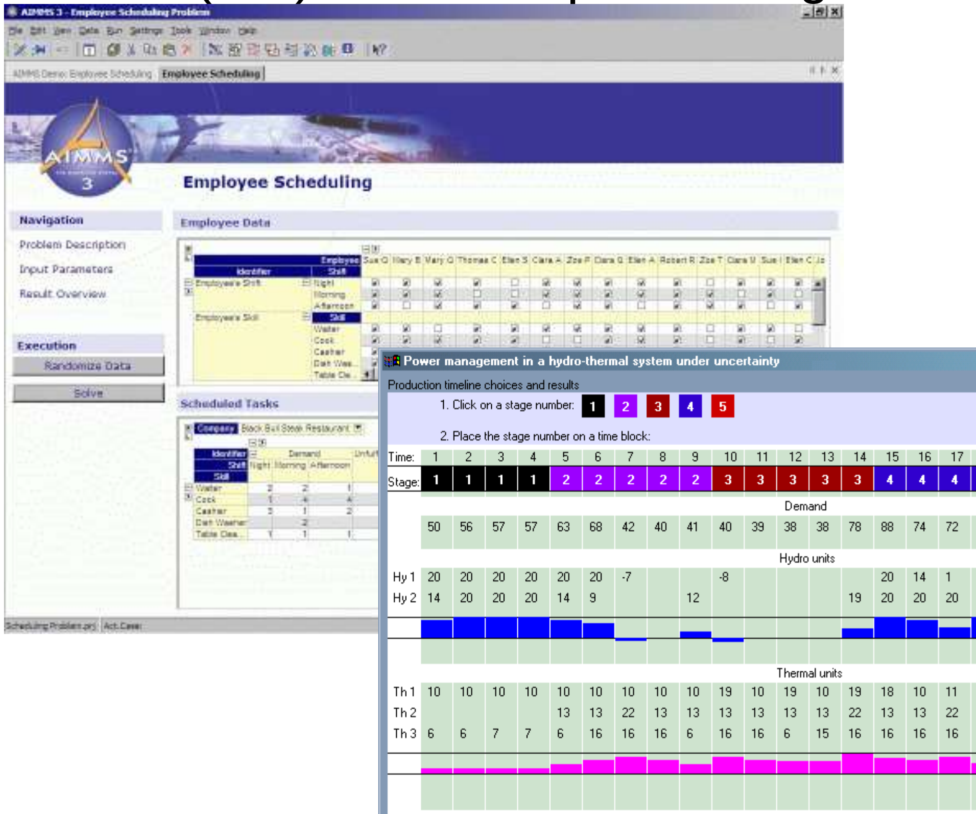
– Add MO-Capability or AML to existing analytical software systems.
– "large" user base, e.g. MATLAB, or SAS



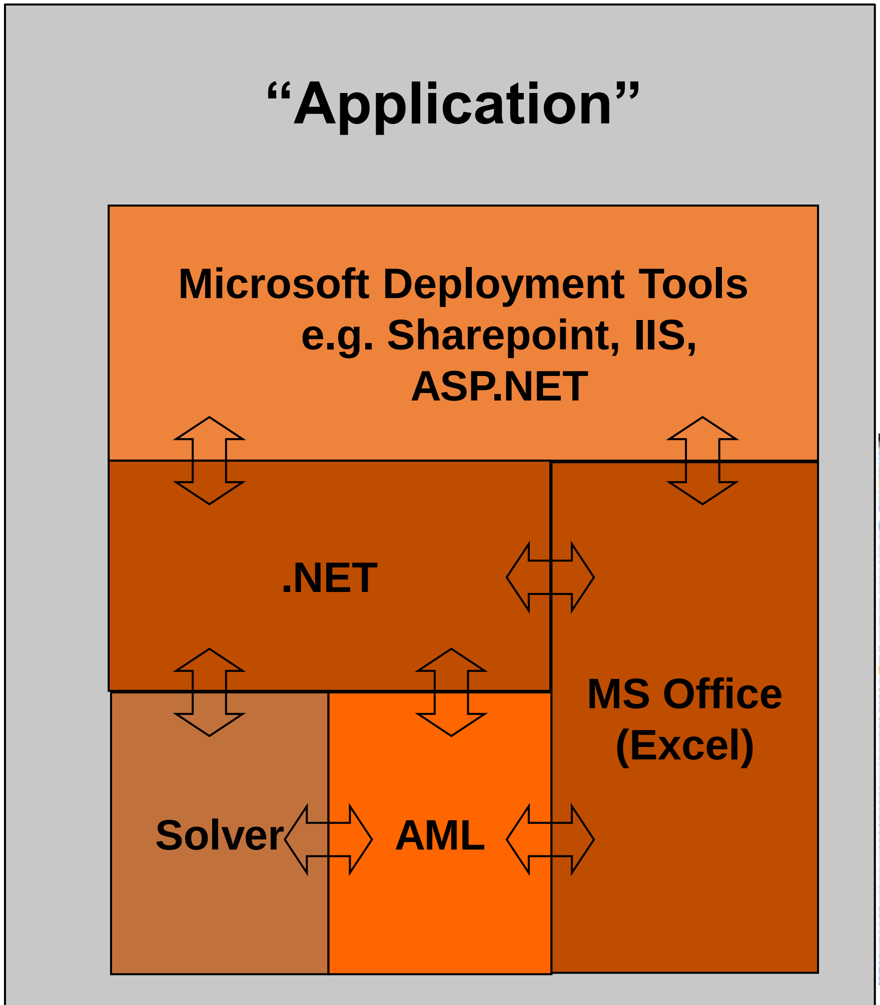
Example – All in One – Bottom Up



- Integrate GUI-builder into AML System
- "small" user base, e.g. AIMMS (Pro) or FICO Xpress-Insight

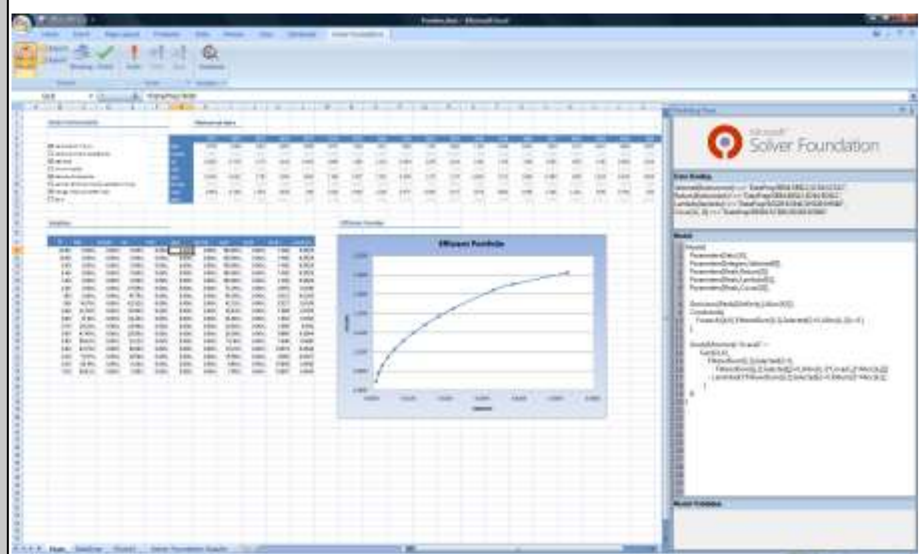


Example – Embedding – Top Down

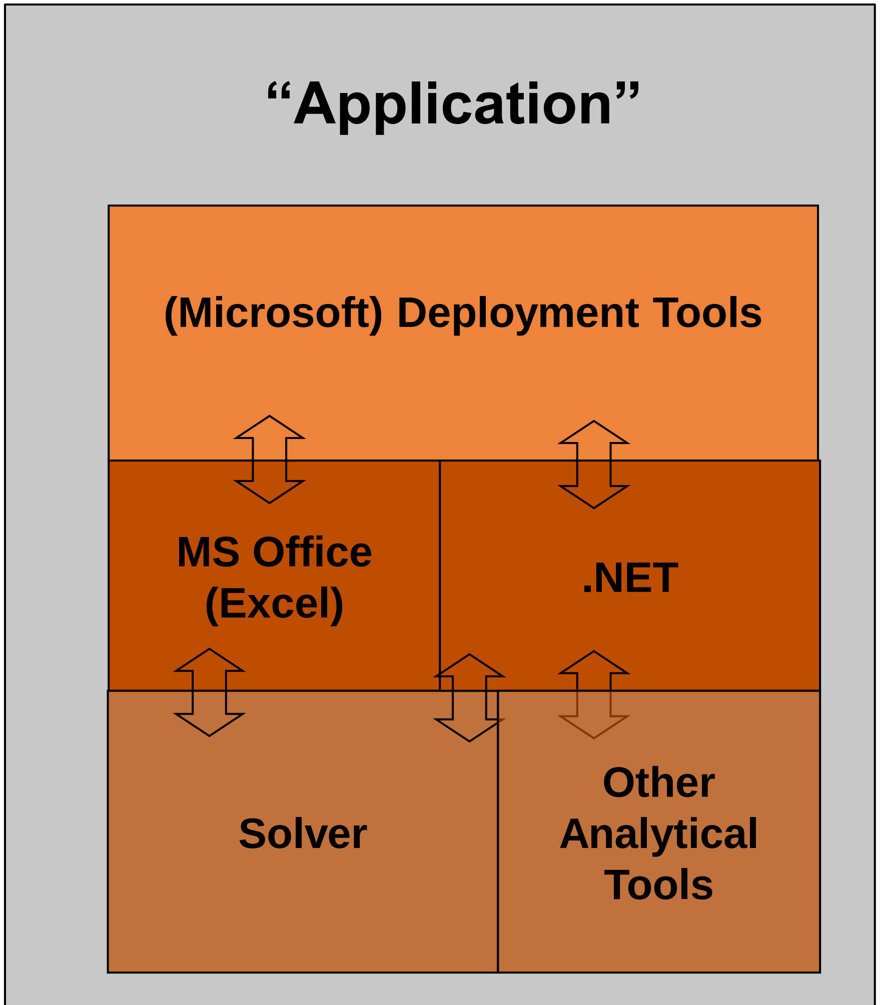


Microsoft Solver Foundation

- . NET based optimization platform
- Includes a declarative AML
- Excel plug-in
- Future??

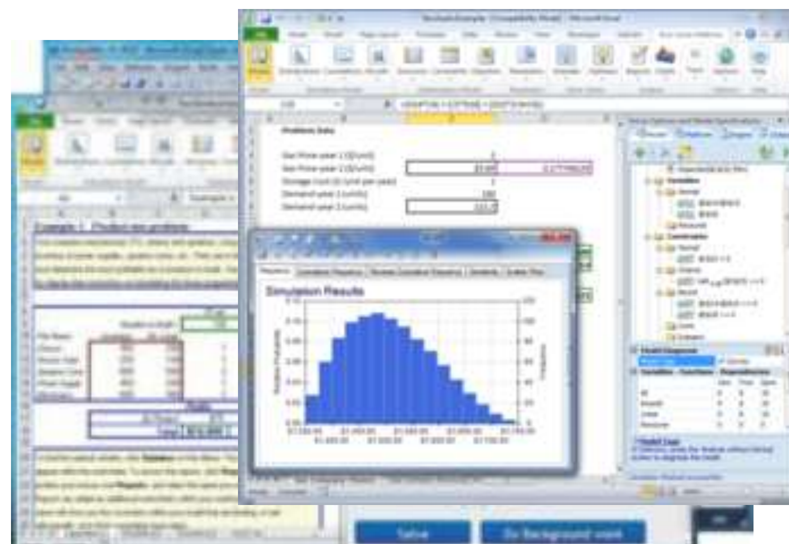


Example – Embedding – Bottom Up



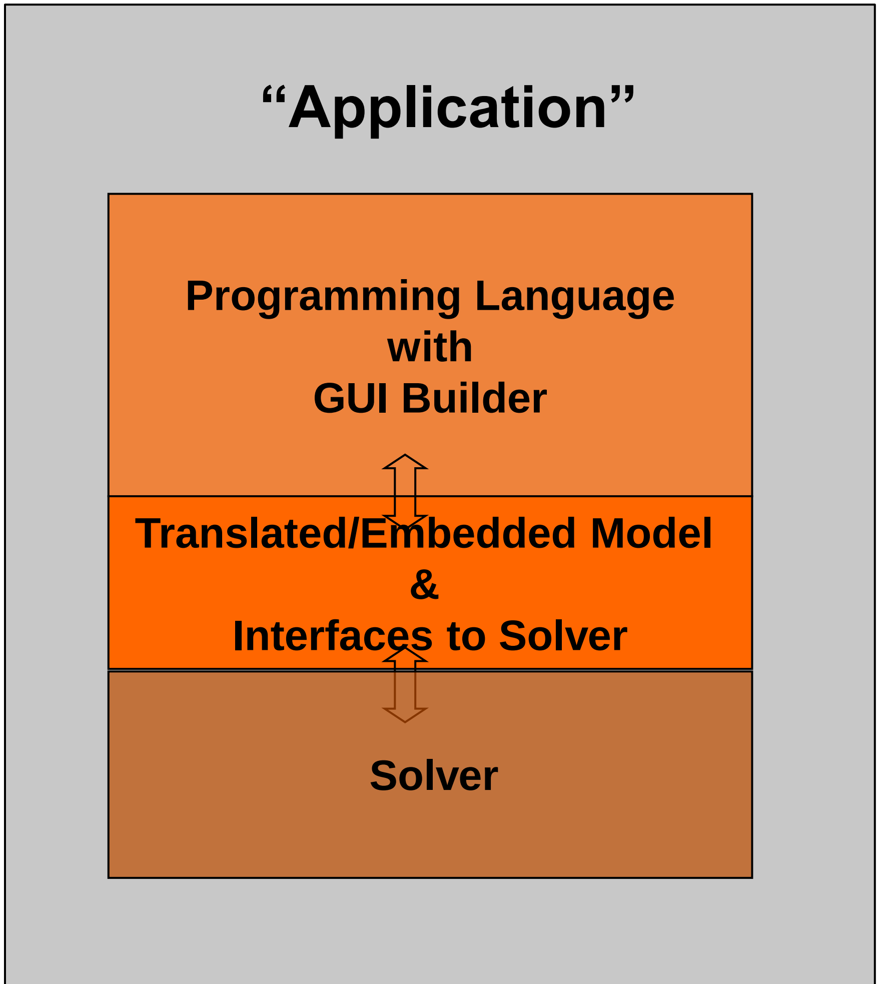
FrontlineSolvers
Developers of the Excel Solver

- Optimization capability included in wide-spread tool
- Excel (and other MS Tools) as deployment environment
- Also other interfaces (e.g. .NET)



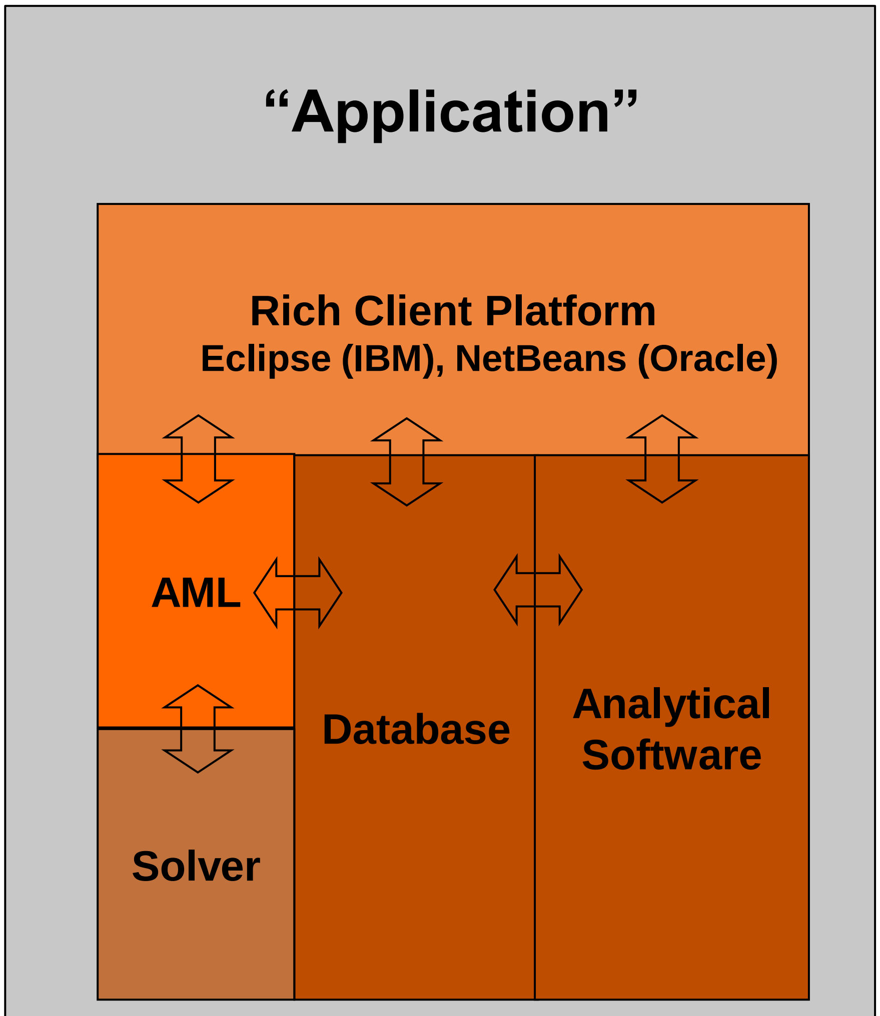
GAMS

Example – Model Translation/Embedding



- **Automatic** translation: early versions of OPL translated OPL source & scripts to C++
- **Manual** translation of model into programming language using tools that mirror the algebraic representation of a single model instance, e.g. IBM ILOG Concert
  - Restricted to linear models
  - Complex models may be difficult to translate

Example – Rich Client Platform

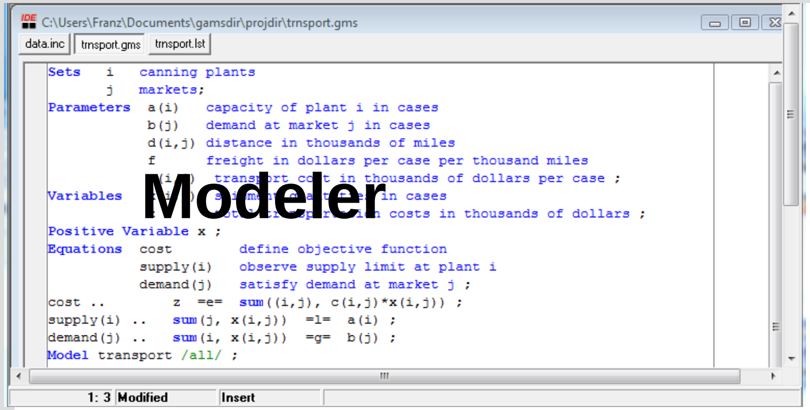


- "Construction Kit" with different connected elements
- Use (open source) existing framework to build applications, e.g. IBM ODM Enterprise

Deploying GAMS Models

- **Integrate GAMS Models**
- Provide Links
- Collaborate

Bridging The Gap



Modeler

Application Developer

GAMS OO API

Concept: Separation of Tasks

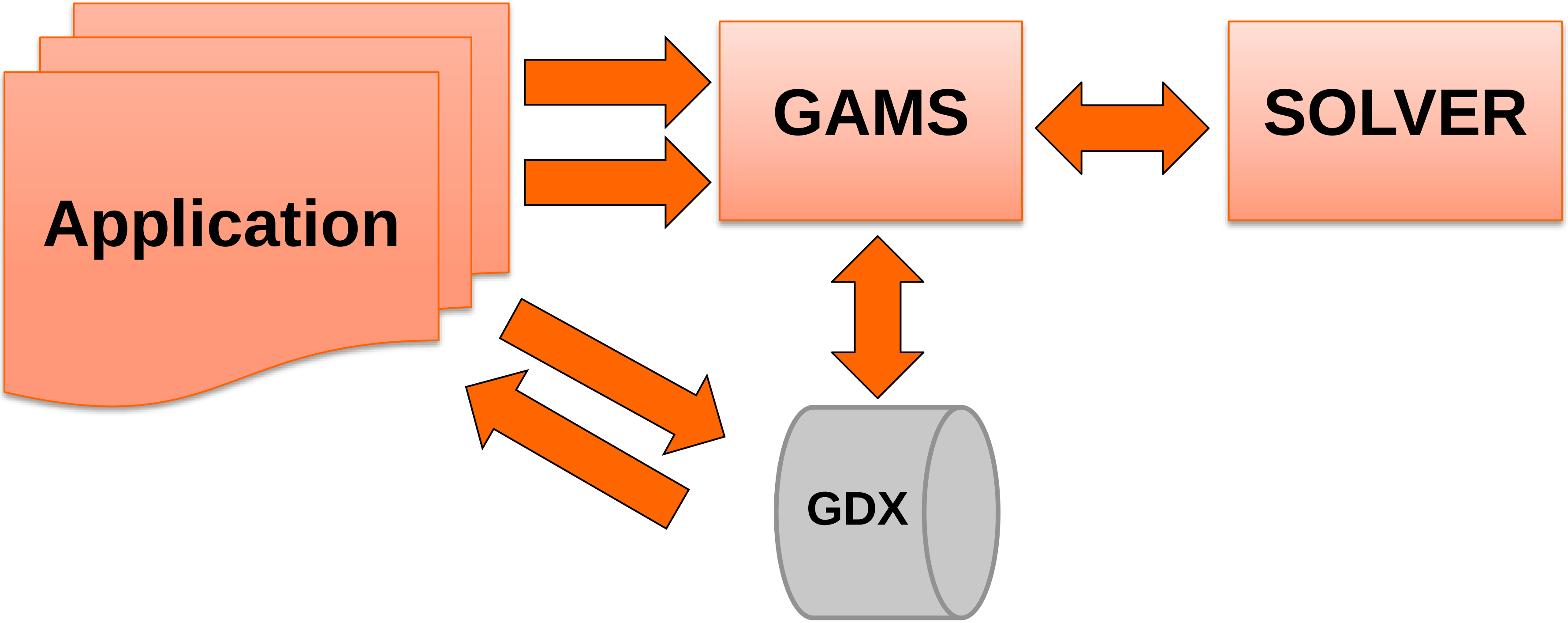
- Use GAMS for modeling and optimization tasks

- Programming languages like C# (.NET), Java and Python are well-suited for developing applications (GUI, Web, …)
- Frameworks available for a wide range of specific tasks, e.g GUI and Web development

- Communication between GAMS and application through GAMS APIs

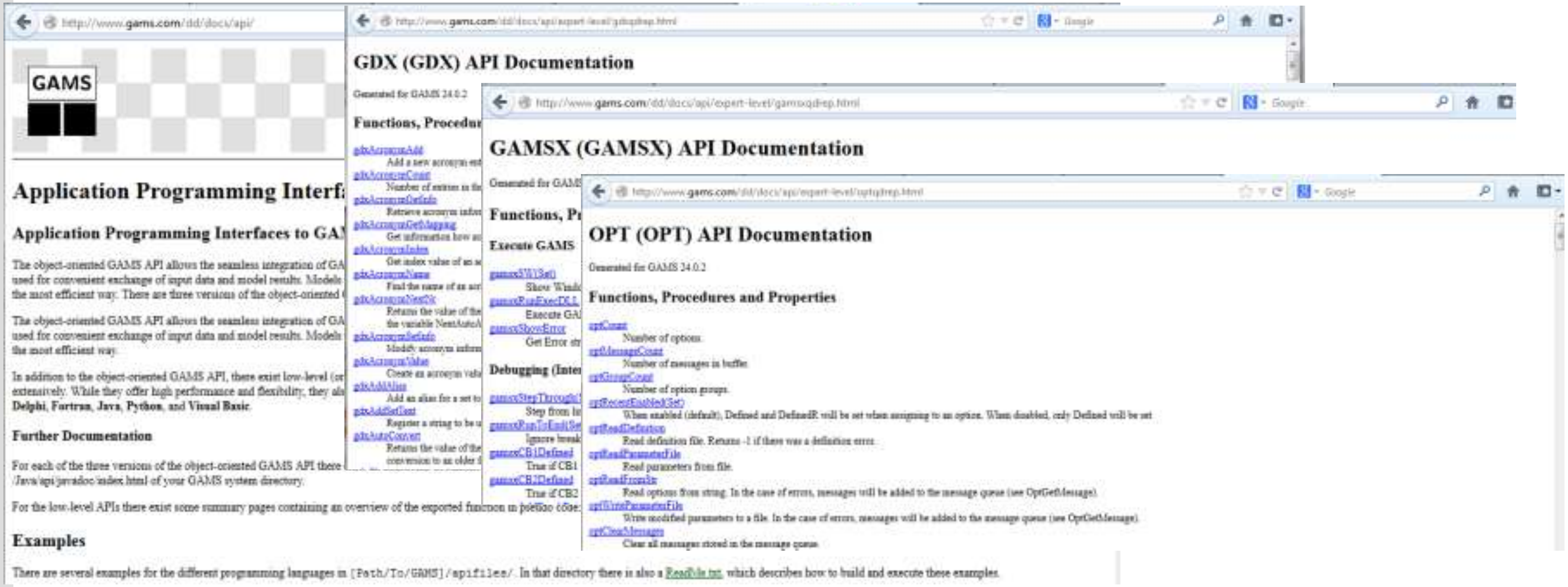
GAMS API: Basic Functionality

1. GDX API: Create Input for GAMS Model
2. Callout to GAMS
   - Option API: GAMS option settings
   - GAMSX API: Start GAMS
3. GDX API: Get Solution from GAMS Model



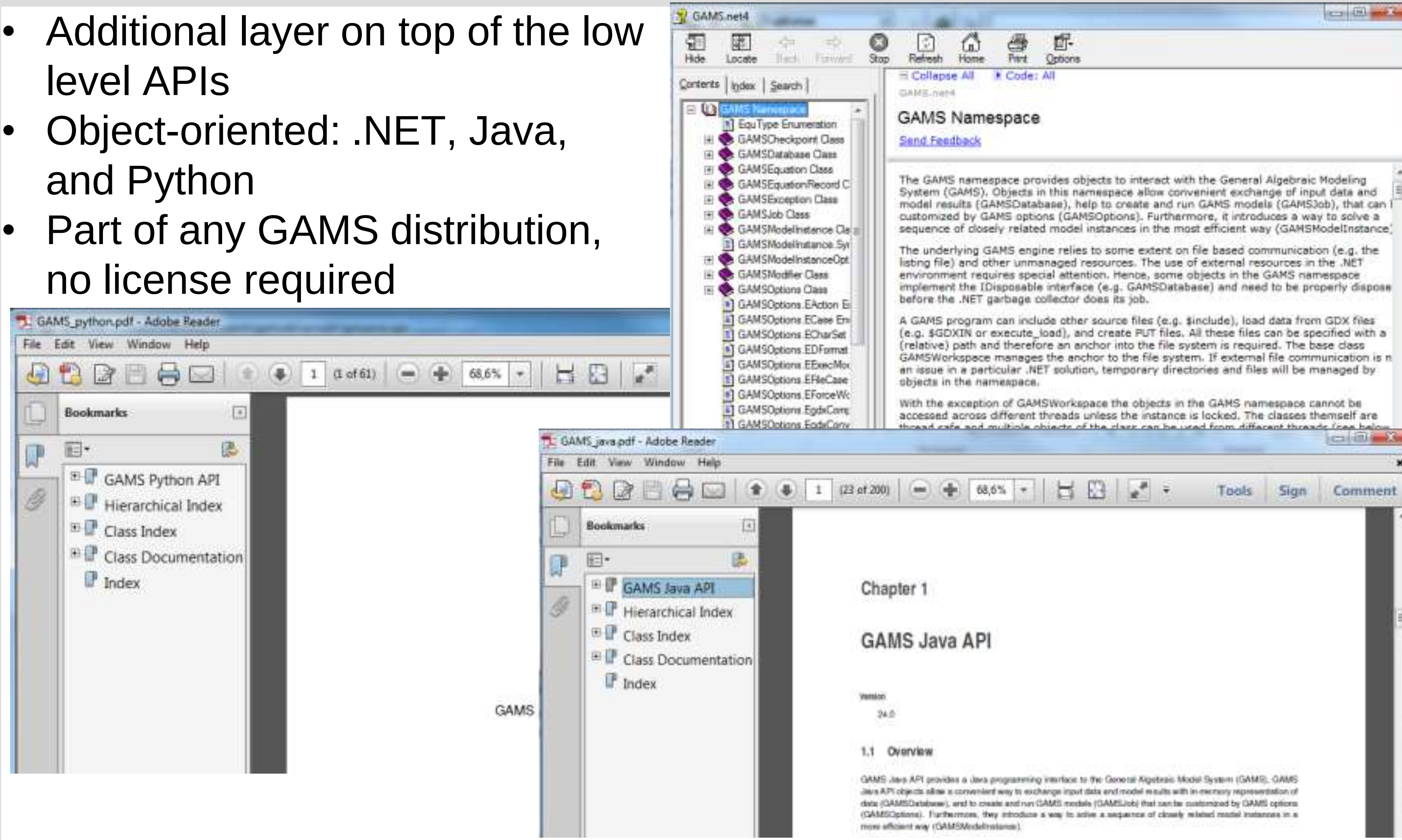
Low level GAMS APIs

- High performance and flexibility
- Automatically generated APIs for several languages (C, Delphi, Java, Python, C#, …)
- Data Exchange (GDX), job control (GAMSX), options (OPT)
- Part of any GAMS distribution, no license required



Object-oriented GAMS API

- Additional layer on top of the low level APIs
- Object-oriented: .NET, Java, and Python
- Part of any GAMS distribution, no license required



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 Class*
- Set GAMS and Solver Options → *GAMSOptions Class*
- Control GAMS execution → *GAMSJob Class*
- Scenario Solving: Feature to solve multiple very similar models in a dynamic and efficient way →*GAMSModelInstance Class*

Small Example - C#

Transport.cs

```
using System;
using GAMS;

namespace TransportSeq
{
    class Transport1
    {
        static void Main(string[] args)
        {
            GAMSWorkspace ws = new GAMSWorkspace();
            GAMSJob t1 = ws.AddJobFromString(GetModelText())

            t1.Run();
            foreach (GAMSVariableRecord rec in t1.OutDB.GetVariable("x"))
            {
                Console.WriteLine("x(" + rec.Keys[0] + ","
                                       + rec.Keys[1] + "):");
                Console.WriteLine("    level=" + rec.Level);
                Console.WriteLine("    marginal=" + rec.Marginal);
            }
        }
```

GAMS

Small Example - C#

```
        static String GetModelText()
        {
            String model = @"
          Sets
               i   canning plants   / seattle, san-diego /
               j   markets          / new-york, chicago, topeka / ;
          Parameters
               a(i)  capacity of plant i in cases
                 /    seattle     350
                      san-diego   600   /
               b(j)  demand at market j in cases
                 /    new-york    325
                      chicago     300
                      topeka      275   / ;

               < . . . >

          Solve transport using lp minimizing z ;";

            return model;
        }
    }
}
```

Small Example - Java

Transport.java

```java
package TransportSeq;
import com.gams.api.*
class Transport1
{
    static void main(String[] args)
    {
        GAMSWorkspace ws = new GAMSWorkspace();

        GAMSJob t1 = ws.addJobFromString(getModelText());
        t1.run();

        for (GAMSVariableRecord rec : t1.OutDB().getVariable("x"))
        {
            System.out.println("x(" + rec.getKeys()[0] + ", "
                                    + rec.getKeys()[1] + "):");
            System.out.println("     level    =" + rec.getLevel());
            System.out.println("     marginal =" + rec.getMarginal());
        }
    }
}
```

Small Example - Python

transport.py

```
from gams import *

if __name__ == "__main__":
    ws = GamsWorkspace()

    t1 = ws.add_job_from_string(get_model_text())
    t1.run()

    for rec in t1.out_db["x"]:
        print rec
```

Scenario Solving - Loop

```
Loop(s,
    f = ff(s);
    solve mymodel min z using lp;
    objrep(s) = z.l;
);
```

- Data exchange between solves possible
- Model rim can change
- Each solve needs to regenerate the model
- User updates GAMS Symbols instead of matrix coefficients

Scenario Solving - GAMSModelInstance

```
foreach (string s in scen)
{
    f.FirstRecord().Value = v[s];
    modelInstance.Solve();
    objrep[s] = z.FirstRecord().Level;
}
```

- Saves model generation and solver setup time
- Hot start (keeps the model hot inside the solver and uses solver’s best update and restart mechanism)
- Data exchange between solves possible
- Model rim unchanged from scenario to scenario

GAMS

GAMSModelInstance etc.

GAMSJob

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

creates

GAMSCheckpoint

- Captures the state of a GAMSJob

initializes

GAMSModelInstance

- A single mathematical model generated by a GAMS solve statement

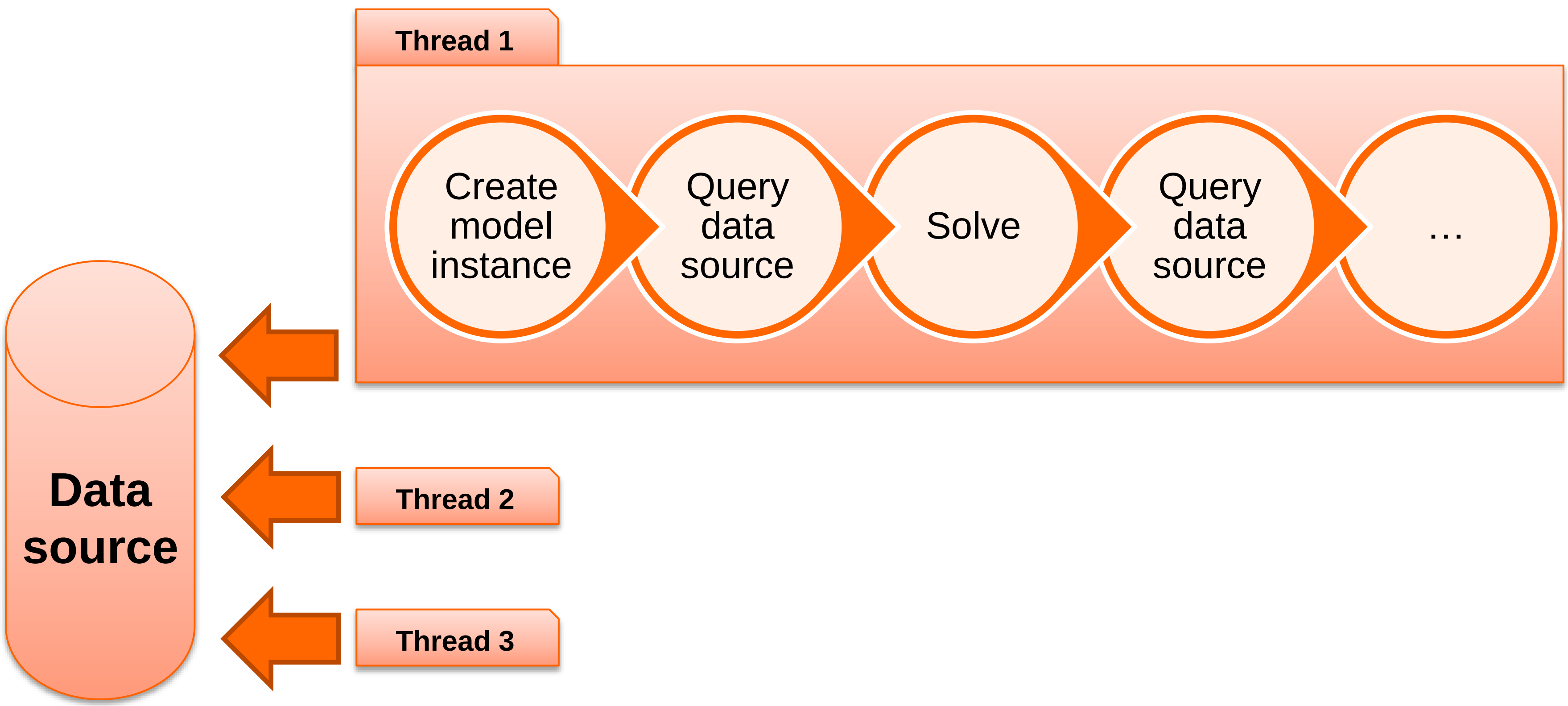
modifies

GAMSModifier

- Marks elements of a GAMSModelInstance to be modifiable

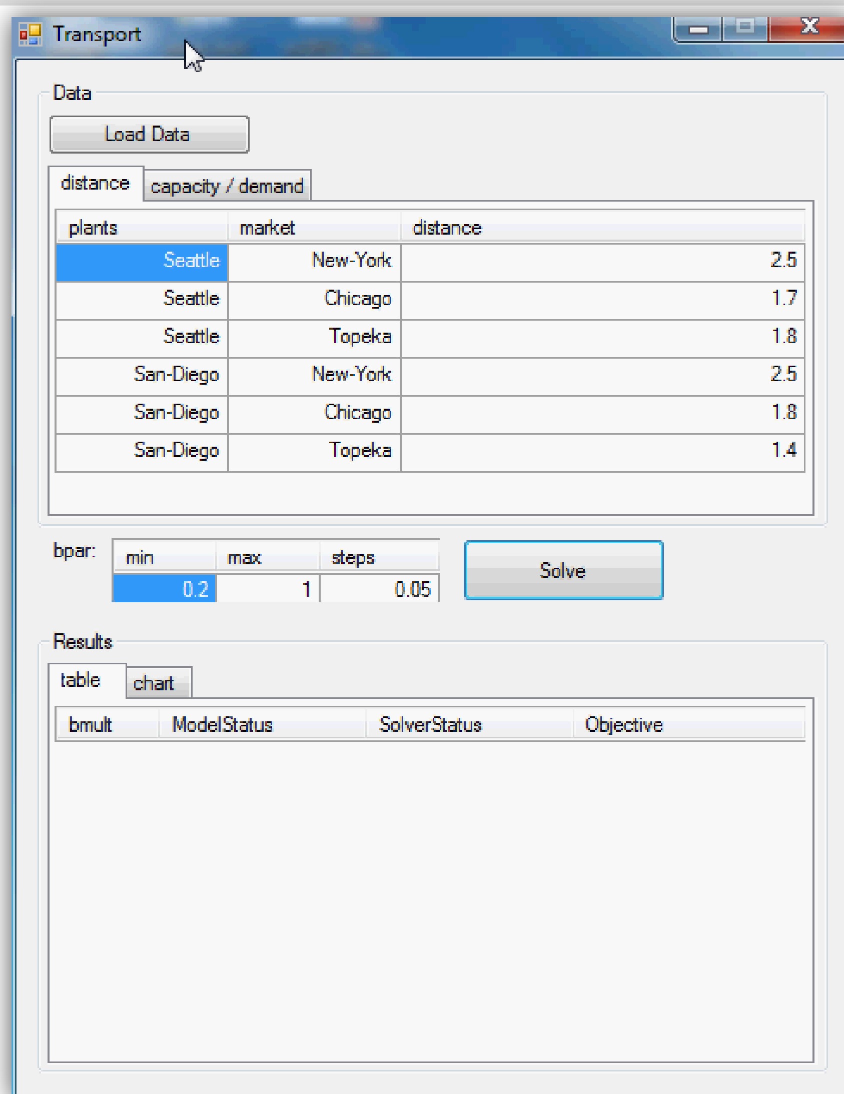
GAMSModelInstances in Parallel

Multiple GAMSModelInstances running in parallel with one common data source:



.NET - Application

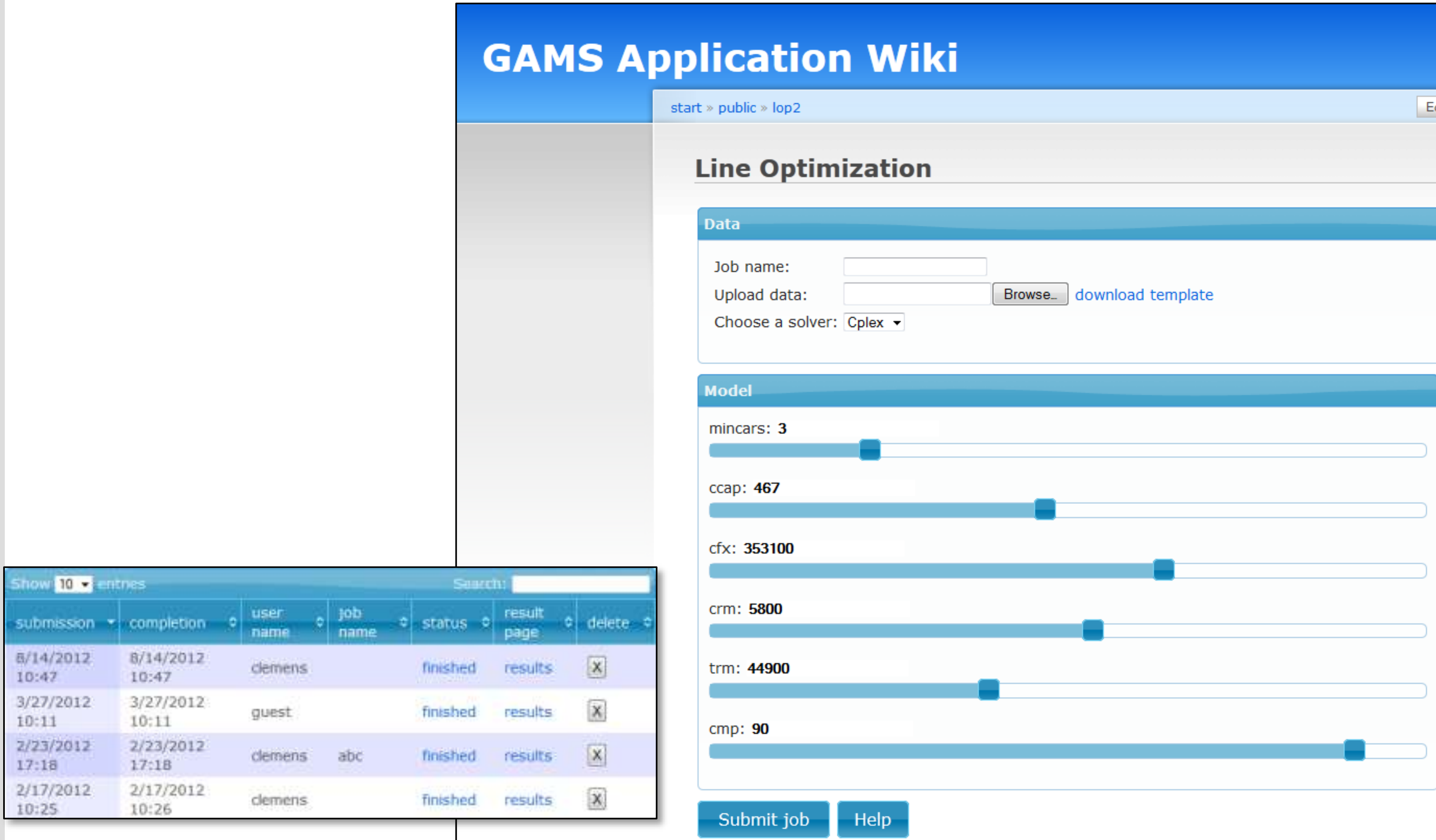
- Example: Small transport desktop application written in C#
- Convenient data preparation
- Representation of the results in a predefined way
- Modeling details are hidden from the user



Wiki as Deployment Environment

- Technologies: Dokuwiki, JavaScript, PHP, …
- Use GAMS for modeling and optimization
- Run GAMS from Wiki environment
- Share results of a GAMS model
- Start a GAMS model asynchronously and come back later to see the results (batch job)
- http://apps.gams.com/doku.php

Wiki – Line Optimization Example



JavaScript & Google Maps

Route 1 | Route 2 | Route 3 | Route 4 | Route 5 | **Route 6** | Route 7 | Route 8
Route 9 | Route 10 | Route 11 | Route 12 | Route 13 | Route 14 | Route 15
Route 16 | Route 17 | Route 18 | Route 19 | Route 20 | Route 21

Search:

from	to	running time	accumulated running time
Amsterdam CS	Apeldoorn	89.00	89.00
Apeldoorn	Hengelo	69.00	158.00
Hengelo	Oldenzaal Grens	18.00	176.00

Showing 1 to 3 of 3 entries

Deploying GAMS Models

Integrate GAMS Models

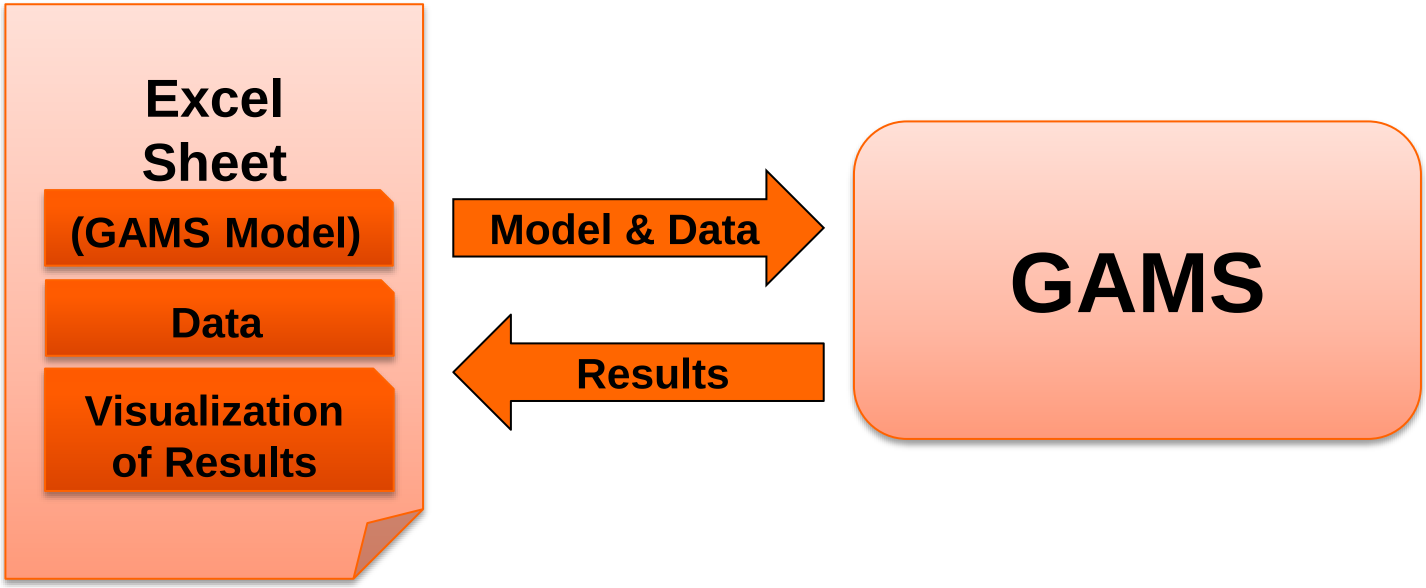
Provide Links

Collaborate

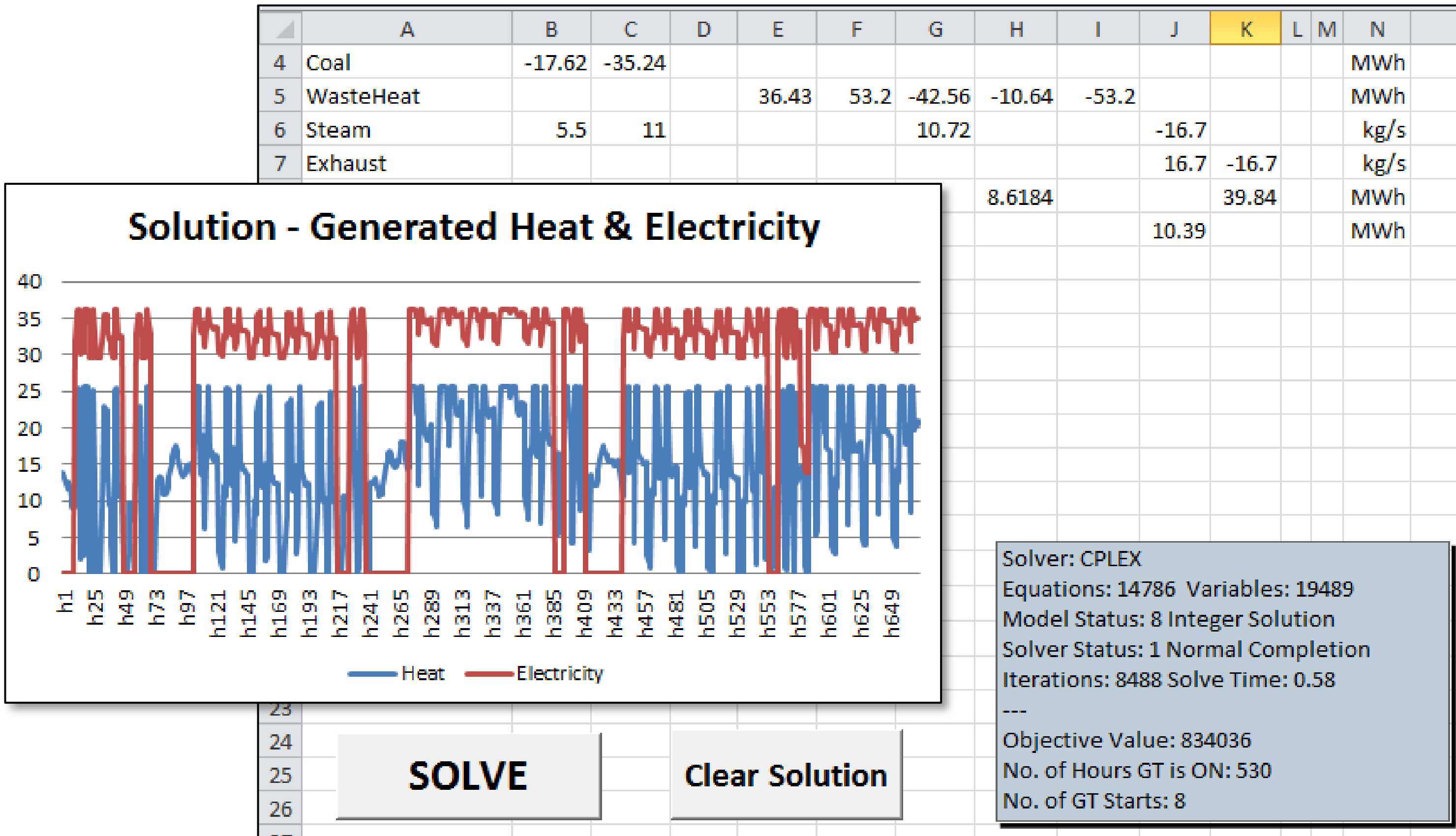
Links to other Software

Example - Excel and GAMS

- VBA GAMS API to call GAMS from Excel
- Exchange of input data and results using either **GDXXRW** or **GDX API**



Excel and GAMS



Excel and Analytics

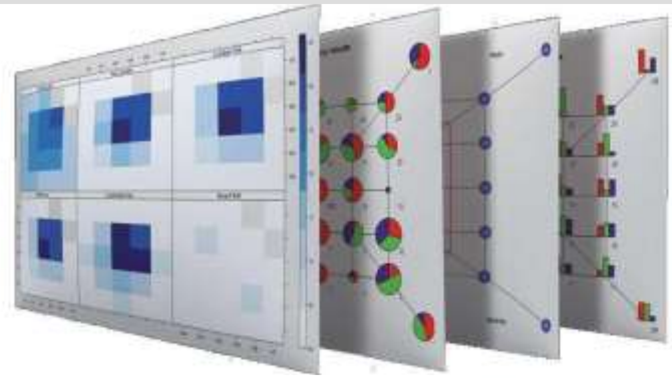
“..the truth is, for many organizations, Excel has always been a part of their business intelligence portfolio for several reasons:

- **Familiarity with Excel..**
- **Built-in flexibility…**
- **Rapid development…**
- **Powerful data connectivity and automation capabilities…**
- **Little to no incremental costs**…

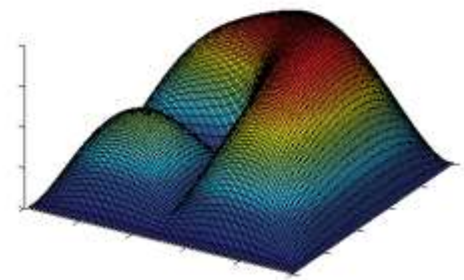
http://tech-book-store.amazon.com/post/TxJQP5UOTREL4J/The-Rise-of-Excel-BI

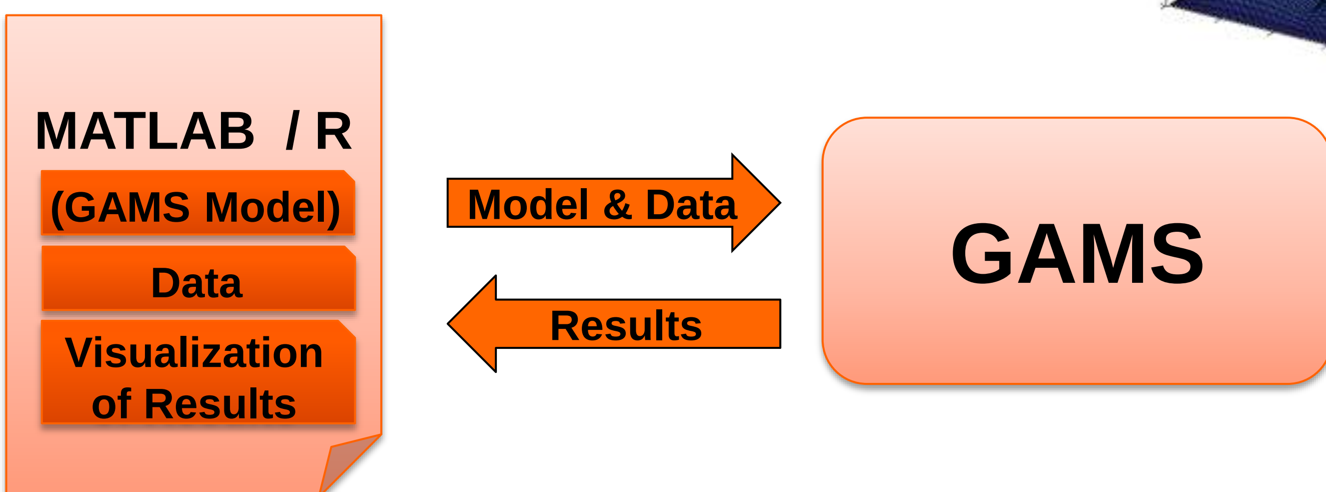
Links to other Software

Example – MATLAB / R and GAMS



- Give MATLAB or R users access to all the optimization capabilities of GAMS
- Allow visualization and analysis of GAMS data directly within MATLAB or R





Deploying GAMS Models

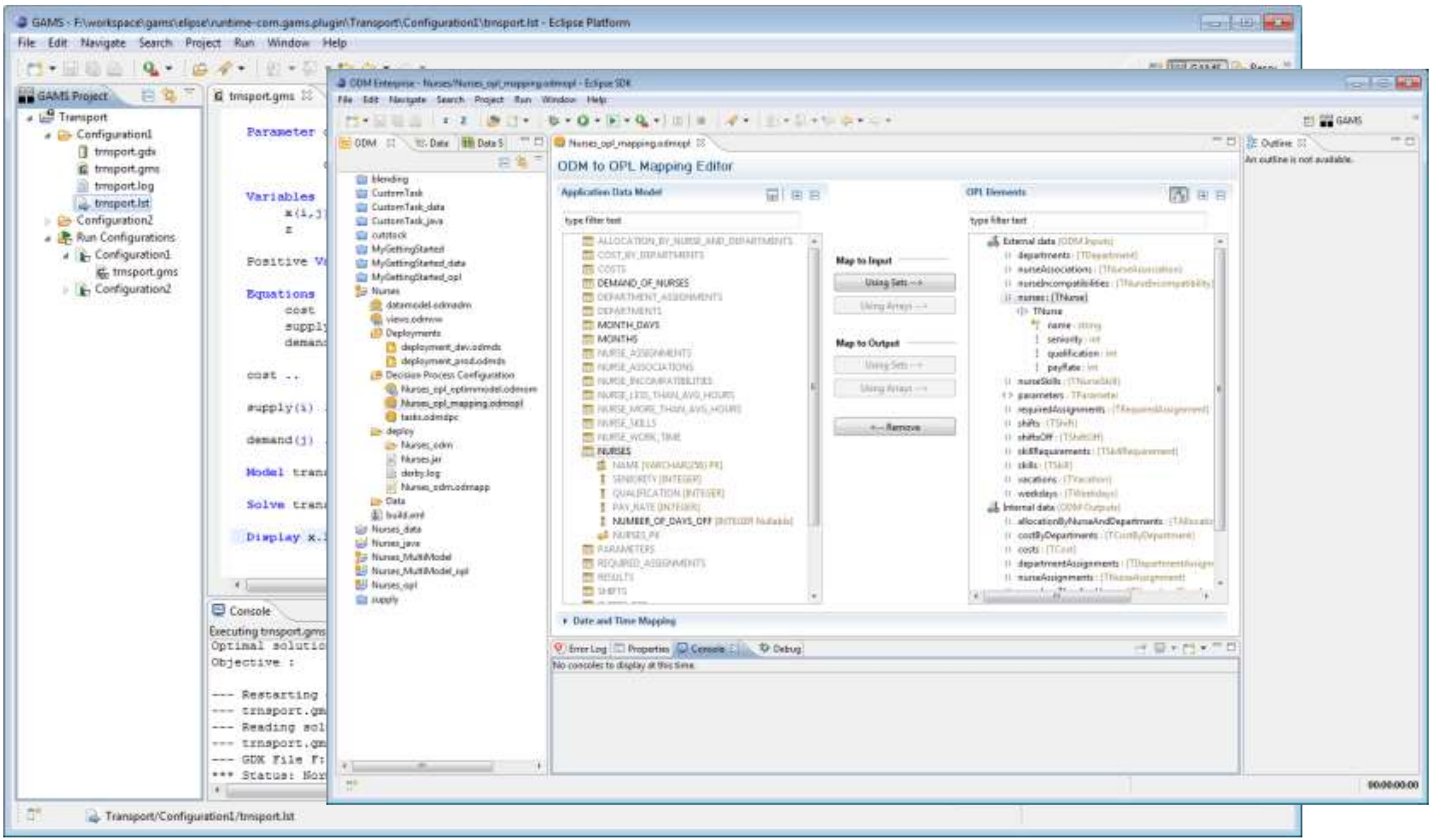
Integrate GAMS Models

Provide Links

Collaborate

Collaborative System Design

GAMS available in IBM ODM Enterprise (Eclipse Plug-in)



GAMS

Recent Enhancements

- GAMS System
- Platforms
- Solvers
- Interfaces
- (Scenario Solver)

GAMS System

- Support for user-defined:
  - Macros
  - Function libraries
- Asynchronous job execution
- Scenario Solver (GUSS)
- Extended Mathematical Programming (EMP) Framework

Excursus: EMP, what?

With new modeling and solution concepts do not:

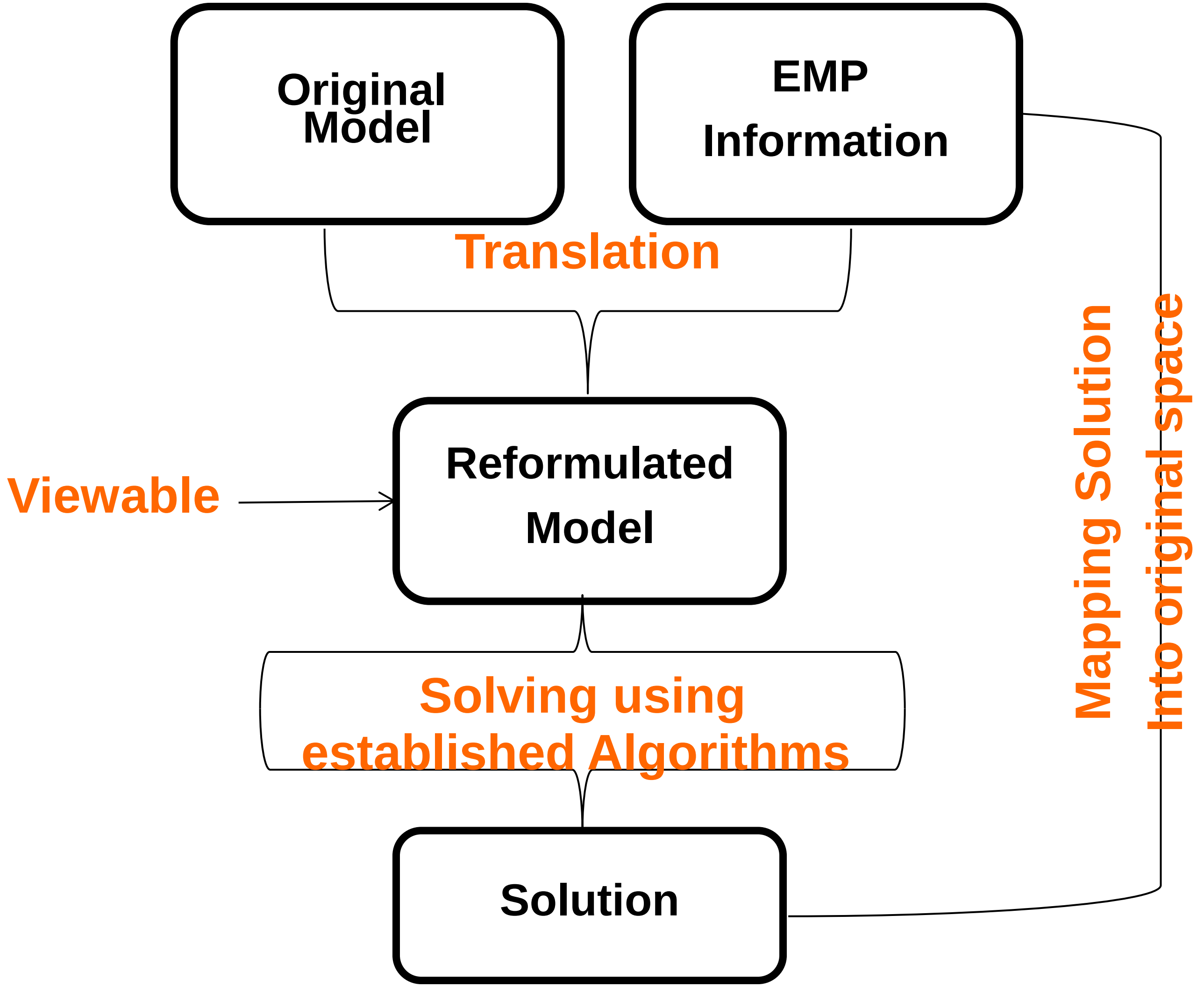
- overload existing GAMS notation right away !
- attempt to build new solvers right away !

But:

- Use existing language features to specify additional model features, structure, and semantics
- Express extended model in symbolic (source) form and apply existing modeling/solution technology
- Package new tools with the production system

→ **Extended Mathematical Programming (EMP)**

JAMS: a GAMS EMP Solver



SP – Deterministic Model

C:\Users\Franz\Documents\gamsdir\projdir\transsp.gms

transsp.gms

```
Scalar f  freight in dollars per case per thousand miles  /90 /
       p  penalty for unsatisfied demand                  / 1 /
       bf demand factor                                   / 1 /;

Parameter c(i,j)  transport cost in thousands of dollars per case ;

          c(i,j) = f * d(i,j) / 1000 ;

display c;

Variables
     x(i,j)   shipment quantities in cases
     u(j)     unsatisfied demand (recourse variable)
     z        total transportation costs in thousands of dollars ;

Positive Variable x,u ;

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)) + p*sum(j,u(j));

supply(i) ..   sum(j, x(i,j))   =l=  a(i) ;

demand(j) ..   sum(i, x(i,j))   =g=  bf*b(j) - u(j) ;

Model transport /all/ ;
```

Example – EMP File & Results

C:\Users\Franz\Documents\gamsdir\projdir\transsp.gms

transsp.gms | transsp.lst

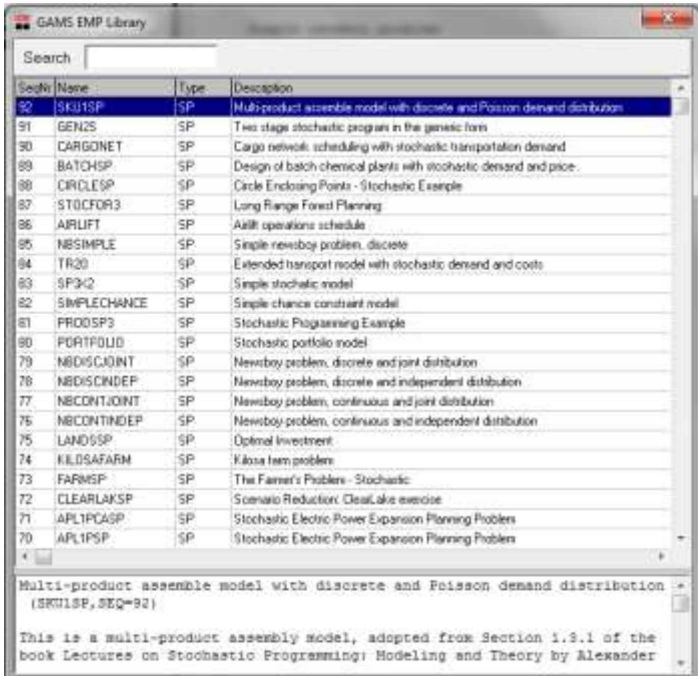
```
file emp / '%emp.info%' /; put emp '* problem %gams.i%'/;
$onput
randvar bf discrete 0.3 0.95
                    0.5 1.00
                    0.2 1.05
stage 2 bf u demand
$offput
putclose emp;

Set scen         scenarios / l,m,h /;
Parameter
    s_bf(scen)     demand factor realization by scenari
    s_u(scen,j)
    s_x(scen,i,j) shipment per scenario
    s_s(scen) ;

Set dict / scen .scenario.''
           bf   .randvar .s_bf
           u    .level   .s_u
           x    .level   .s_x /;

Solve transport using emp minimizing z scenario dict;

Display s_bf, s_x, s_u;
```

\transsp.lst

```
----     87 PARAMETER s_bf  demand factor realization by scenario

l 0.950,    m 1.000,    h 1.050


----     87 PARAMETER s_x  shipment per scenario

               new-york     chicago      topeka

l.seattle        35.000     315.000
l.san-diego     290.000                 288.750
m.seattle        35.000     315.000
m.san-diego     290.000                 288.750
h.seattle        35.000     315.000
h.san-diego     290.000                 288.750


----     87 PARAMETER s_u

     new-york

h      16.250


**** REPORT FILE SUMMARY
```

Stochastic Programming in GAMS

- Based on Extended Mathematical Programming (EMP) framework
- Discrete and parametric distribution for random parameters
- Supports multi-stage recourse problems and chance constraint models
- Make it easy to add uncertainty to existing deterministic models:
  - use specialized algorithms (DECIS or LINDO)
  - or create deterministic equivalent (free solver DE)
- SP examples in the GAMS EMP library



Platforms

- Support for MAC OS X
- Cross-platform licenses
- Wine (Linux, Mac)

Solvers

- **GLoMIQO**: Branch-and-bound global optimization for mixed-integer quadratic models

- **Lindo**: Global and stochastic optimization

- **SULUM:** Linear (and soon) mixed integer optimization

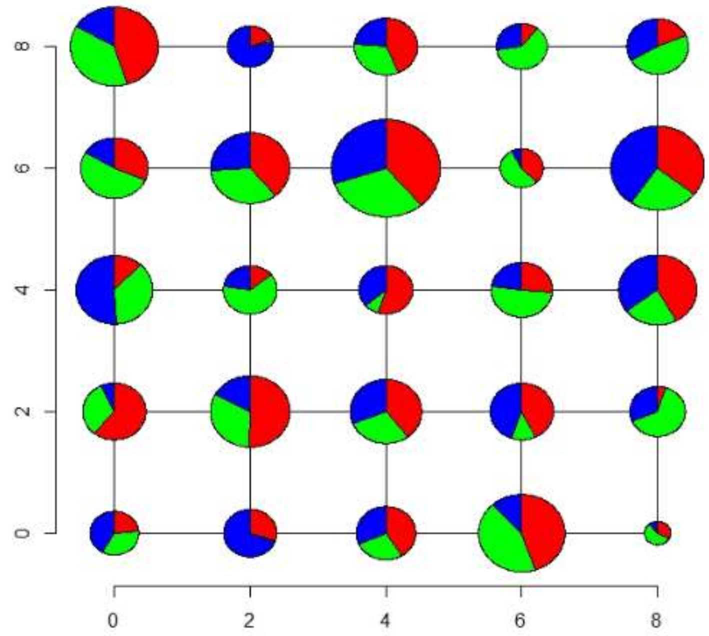
Current GAMS Distribution 24.0.2

Released February 24, 2013 www.gams.com/download

- **Solver updates**
  - BARON 11.9.1
  - CONOPT 3.15I
  - CPLEX 12.5
  - GLOMIQO 2.0
  - GUROBI 5.1
  - KNITRO 8.0
  - LINDO 7.0.1.487
  - MOSEK 6 rev 148
  - XPRESS 23.01.06
  - …

Interfaces

- OO API for .NET, Java and Python
- GDXRRW
  - Connects R-System with GAMS
  - Presents data in a natural form for R users



Source: http://blog.modelworks.ch

GAMS

Recent Enhancements

- GAMS System
- Platforms
- Solvers
- Interfaces
- **Scenario Solver**

GAMS

Solving Data Related Models

The common way:

```
Set s / s1*s10 /
Parameter
    A_s(s,i,j) Scenario data
    xlo_s(s,j) Scenario lower bound for variable x
    xl_s(s,j)  Scenario solution for x.l
    em_s(s,i)  Scenario solution for e.m;
Loop(s,
  A(i,j) = A_s(s,i,j);
  x.lo(j)= xlo_s(s,j);
  solve mymodel min z using lp;
  xl_s(s,j) = x.l(j);
  em_s(s,i) = e.m(i);
);
```

- GAMS generates model and writes it to disk
- GAMS writes database to scratch files on disk
- GAMS calls solver and vacates memory
- After solver is done: GAMS restarts and swaps database

Gather-Solve-Update-Scatter (GUSS)

Idea:

- Solve one particular model instance (i.e. same individual variables and constraints) with different data and bounds
- Works with any model type in GAMS
- Change in model data for a subsequent solve statement does not depend on previous model solutions

Advantages:

- GAMS stays live
- Model generation only once
- Fast communication with solver
- Hotstarts and advanced basis information

GAMS

GUSS: Example

```
Set s / s1*s10 /
Parameter
    A_s(s,i,j) Scenario data
    xlo_s(s,j) Scenario lower bound for variable x
    xl_s(s,j)  Scenario solution for x.l
    em_s(s,i)  Scenario solution for e.m;
Set dict / s.   scenario. ''
           A.   param.     A_s
           x.   lower.     xlo_s
           x.   level.     xl_s
           e.   marginal.  em_s   /;
solve mymodel min z using lp scenario dict;
```

GUSS: Performance

- Model with 3838 scenarios

Setting	Solve time (secs)
Solvelink=0 (default)	294.36
Solvelink=%Solvelink.CallModule%	239.57
Solvelink=%Solvelink.LoadLibrary%	16.21
GUSS	9.18

GUSS: Example

From Erwin Kalvelagen’s Blog

Implementation	number of MIP models	solve time	rest of algorithm	total time
Traditional GAMS loop (call solver as DLL)	100,000	1068 sec	169 sec	1237 sec
Scenario Solver	100,000	293 sec	166 sec	459 sec

Implementation	number of MIP models	Worker threads	parallel sub-problem time	rest of algorithm (serial)	total time
Parallel + Scenario Solver	100,000	4	116 sec	67 sec	183 sec

http://yetanothermathprogrammingconsultant.blogspot.de/2012/04/parallel-gams-jobs-2.html

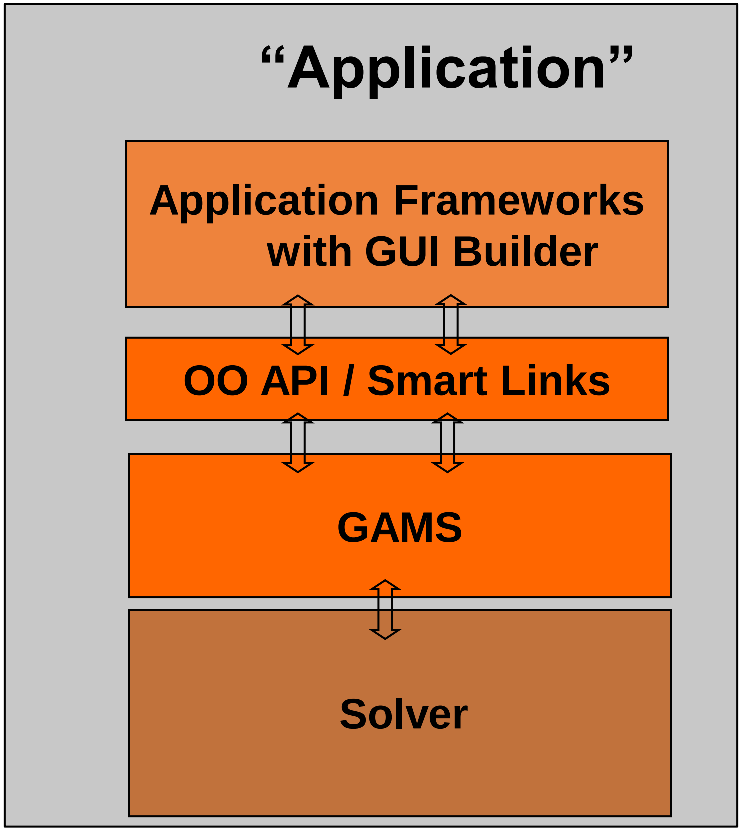
Summary

- Change in focus from algorithm to application
- Different approaches to deploy optimization models
- The challenge remains to provide a deployment environment, which
  - leads to productivity increase
  - establishes confidence in optimization methods
  - and is easy to maintain

Summary

We believe in

- the important role of an AML to build a failsafe and reliable system
- the integration into other systems through
  - OO APIs to existing application frameworks
  - Smart links to other analytical environments



Thank You !

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

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

http://www.gams.com