GAMS

Transportation Model

Michael Bussieck mbussieck@gams.com

Lutz Westermann lwestermann@gams.com

GAMS Development Corporation

www.gams.com

GAMS Software GmbH

www.gams.de

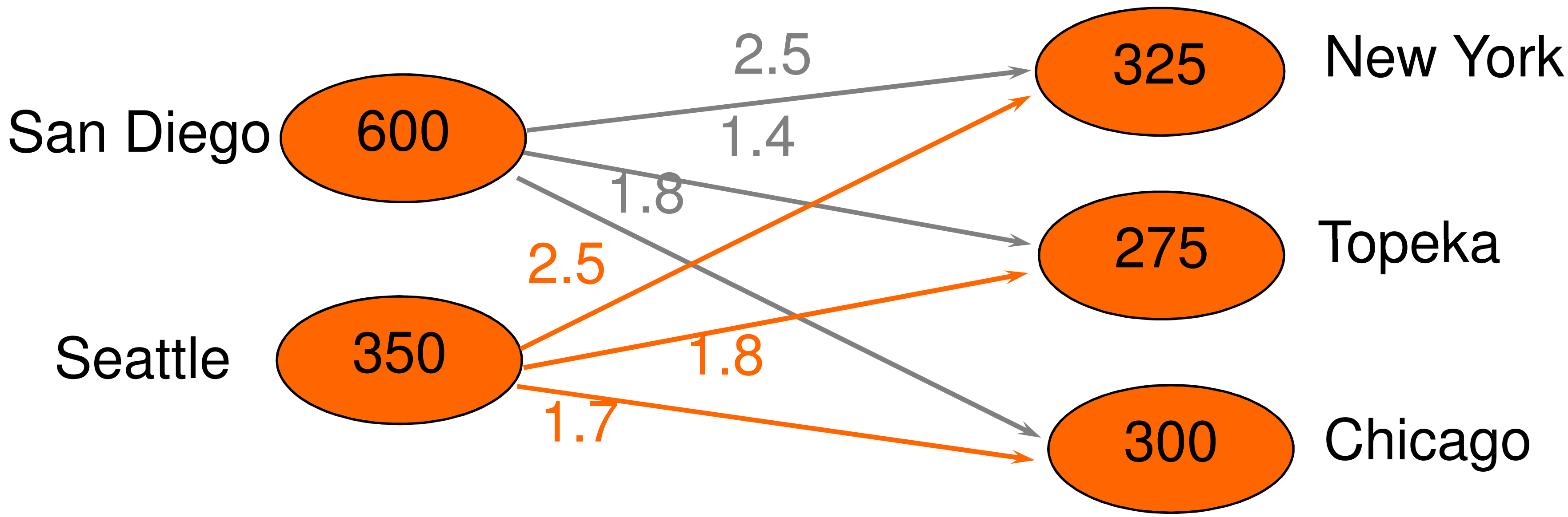
INFORMS 2009 San Diego

What is a Model?

- Mathematical Programming (MP) Model
  - List of Equations

- Collection of several intertwined MP Models
  - Data Preparation
  - Data Calibration
  - “Solution” Module (e.g. sequential, parallel, loop)
  - Report Module

A Transportation Model



Minimize	Transportation cost
subject to	Demand satisfaction at markets
	Supply constraints

Mathematical Algebra

$$
\begin{aligned}
\sum_{\substack{c,p:\\(c,p)\in\mathcal{N}}} tcost \cdot dist(c,p) \cdot x_p^c &\rightarrow \min \\
\sum_{\substack{c,p:\\(c,p)\in\mathcal{N}}} x_p^c &\leq sup(c) \quad \forall c \\
\sum_{\substack{c,p:\\(c,p)\in\mathcal{N}}} x_p^c &\geq dem(p) \quad \forall p \\
x_p^c &\geq 0 \quad \forall c,p:\ (c,p)\in\mathcal{N}
\end{aligned}
$$

GAMS Algebra

```
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/ ;
```

A few Word about GAMS Syntax

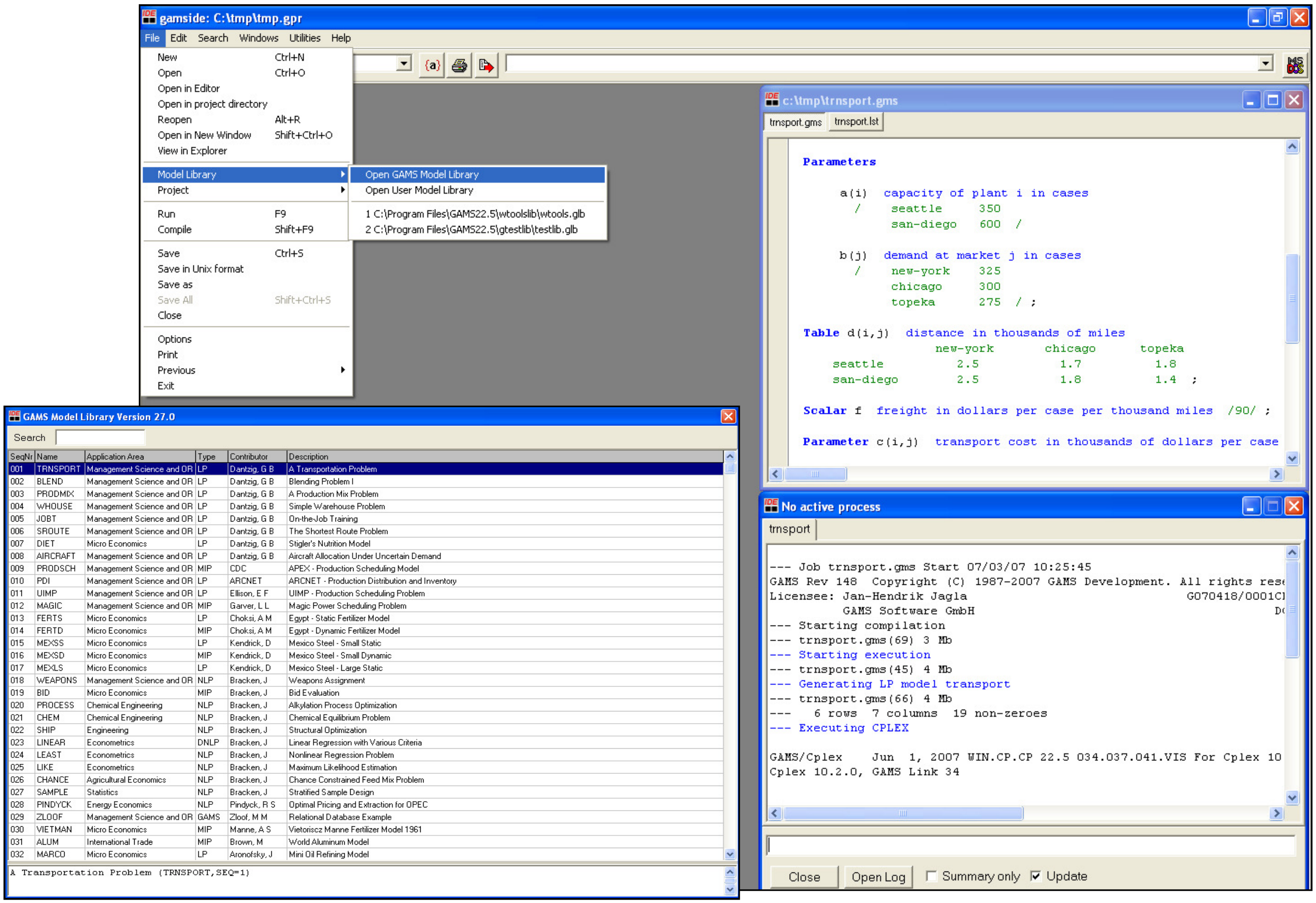
- Symbols:
  - Sets
  - Parameters
  - Variables
  - Equations
  - Models
  - ASCII Output Files

```
Sets          i           canning plants    / seattle, san-diego /;
Parameters    a(i)        capacity of plant i in cases
       /      seattle      350
              san-diego    600   /;
Variables     x(i,j)      shipment quantities in cases;
Equations     supply(i)   observe supply limit at plant i;
Model         transport   /all/ ;
File          fx          some file / 'c:\t\text.txt' /
```

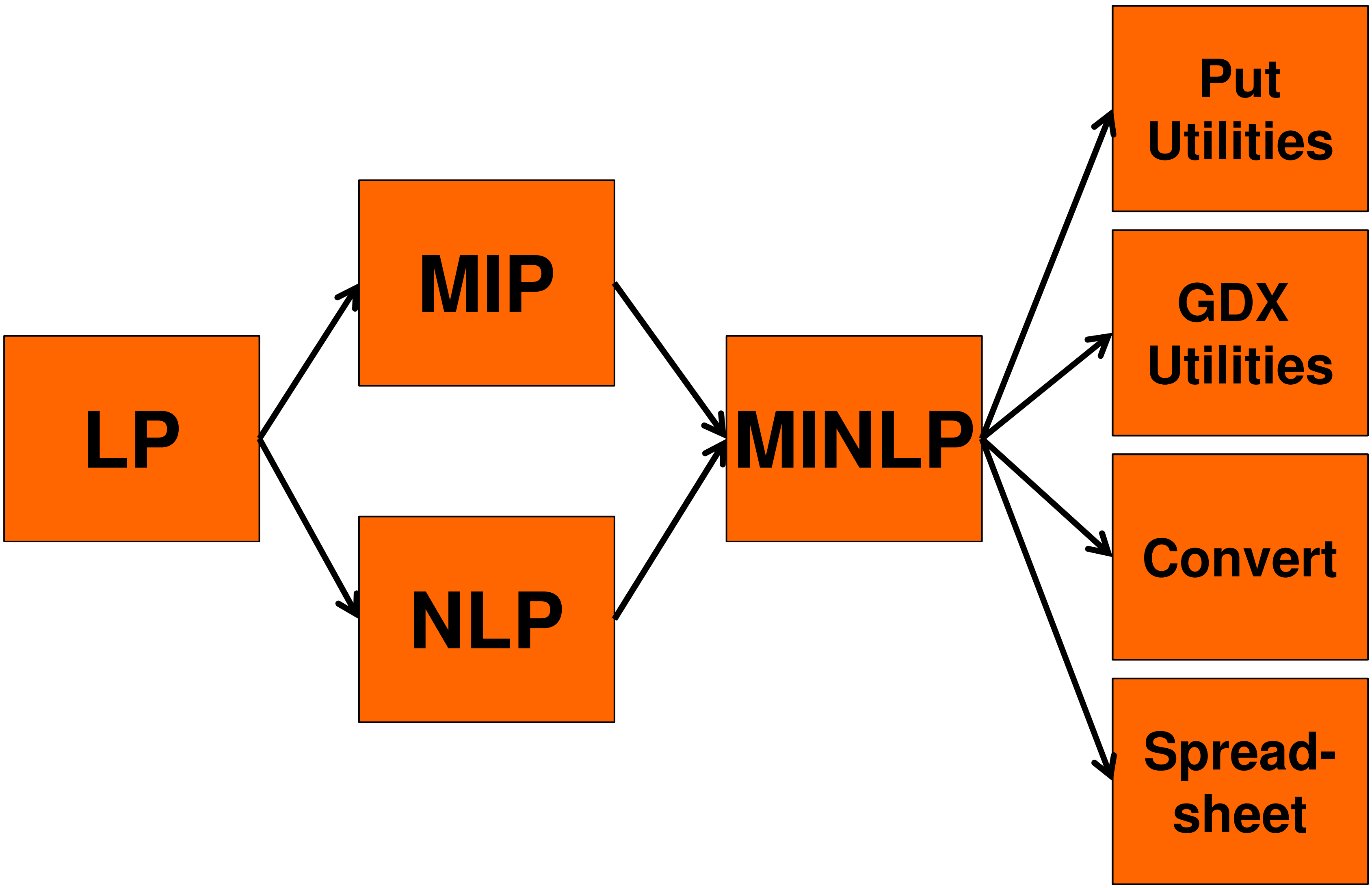
- Statements
  - Declarations
  - Data Assignments
  - Equation Definition
  - Programming Flow Control
  - Option statement

```
Parameter c(i,j);
c(i,j) = f * d(i,j) / 1000 ;
supply(i) ..   sum(j, x(i,j))  =l=  a(i);
loop(i, put fx i.tl);
option reslim=10;
```

Demo! Transportation Model



Modifications to the transport model



Types of Variables

- Continuous Variables
  - Free/Positive/Negative
  - Lower and/or upper bound

- Binary Variables
  - Either 0 or 1

- Integer Variables
  - Any integer number

- Semicont/Semiint Variables
  - 0 or above a given minimum

- Special Ordered Set Variables (SOS1, SOS2)

GAMS

Binary Variables

- Powerful Tool to model yes/no decisions

- Models with discrete variables (MIP)
  - Solved using Branch-and-Cut algorithms (lots of LPs)
  - Theoretically difficult problem class
  - Practical:
    - mixed bag
    - *Art of Modeling*

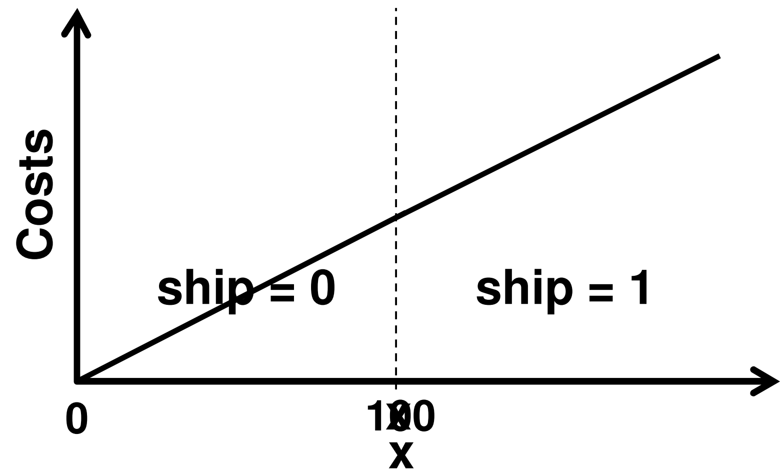
- Example: Minimum Shipment
  - Ship at least 100 tons or don’t ship

Demo! Binary Vars: Minimum Shipment

- Continuous Variable *x* (shipment)

- Binary Variable *ship* (decision whether to ship or not):
  - *ship* = 1 if $x \geq 100$
  - *ship* = 0 if $x = 0$

- Coupling Constraints:
  - $x \geq 100 * ship$
  - $x \leq bigM * ship$



- How big do we have to make bigM?

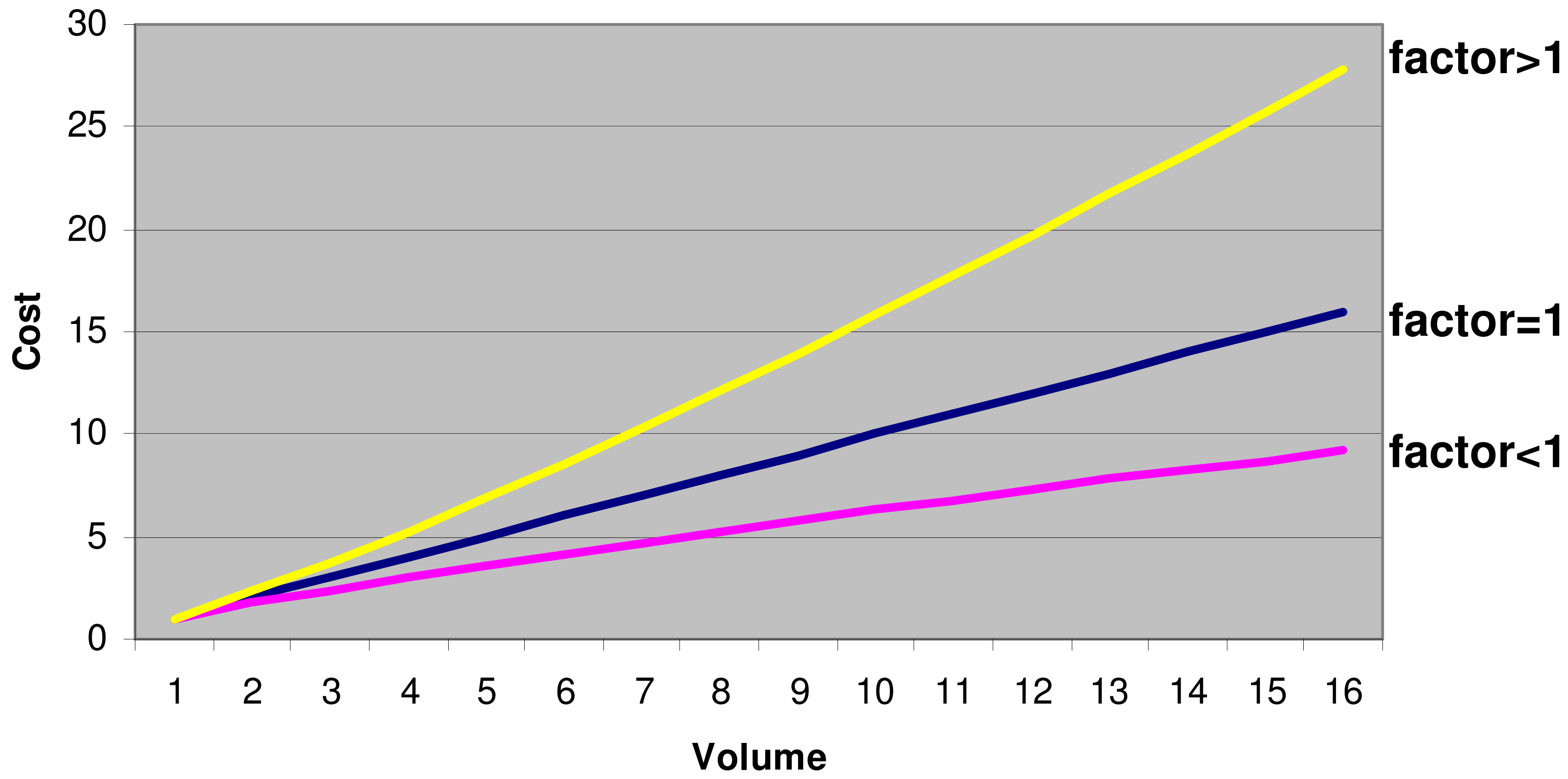
GAMS

Implement Min/Max Shipments (MIP)

```
Parameter rep1(i,j,*)    Shipments between plants and markets
          rep2(*)        Objective value;

rep1(i,j,'lp') = x.l(i,j);
rep2('lp')     = z.l;

Scalars mins / 100 /
        bigm / 325 /;

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=   mins*ship(i,j);
maxship(i,j).. x(i,j) =l=   bigm*ship(i,j);

model m2 min shipments / all /;
solve m2 using mip minimizing z;
rep1(i,j,'mip') = x.l(i,j);
rep2('mip')     = z.l;

option mip=coincbc
solve m2 using mip minimizing z;
rep1(i,j,'mip-coincbc') = x.l(i,j);
rep2('mip-coincbc')     = z.l;
display rep1,rep2;
```

Demo! NL-Model: Economy of Scales

$$Cost = const \cdot Volume^{factor}$$



Implement Nonlinear Cost (NLP)

```
* 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;
rep1(i,j,'nlp-convex') = x.l(i,j);
rep2('nlp-convex')     = z.l;

beta = 0.6;
solve m3 using nlp minimizing z;
rep1(i,j,'nlp-concave') = x.l(i,j);
rep2('nlp-concave')     = z.l;

option nlp=baron;
solve m3 using nlp minimizing z;
rep1(i,j,'nlp-baron') = x.l(i,j);
rep2('nlp-baron')      = z.l;

display rep1,rep2;
```

Implement Min/Max and Nonlinear (MINLP)

```
* min/max and nonlinear objective

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

option minlp=baron;
solve m4 using minlp minimizing z;
rep1(i,j,'minlp-bar') = x.l(i,j);
rep2('minlp-bar')     = z.l;

option minlp=lindoglobal;
solve m4 using minlp minimizing z;
rep1(i,j,'minlp-lin') = x.l(i,j);
rep2('minlp-lin')     = z.l;

display rep1,rep2;
```

Data Connectivity

- Data Import/Export from *Standard Applications*
  - ➢ Text files
  - ➢ Gams Data eXchange (GDX)
    - MS Office, Databases, …
- Capture an *Instance*
  - – Reproducibility of Model/System Bugs
  - – Problems: Life Database/different Platforms
  - ➢ convert
  - ➢ dumpopt

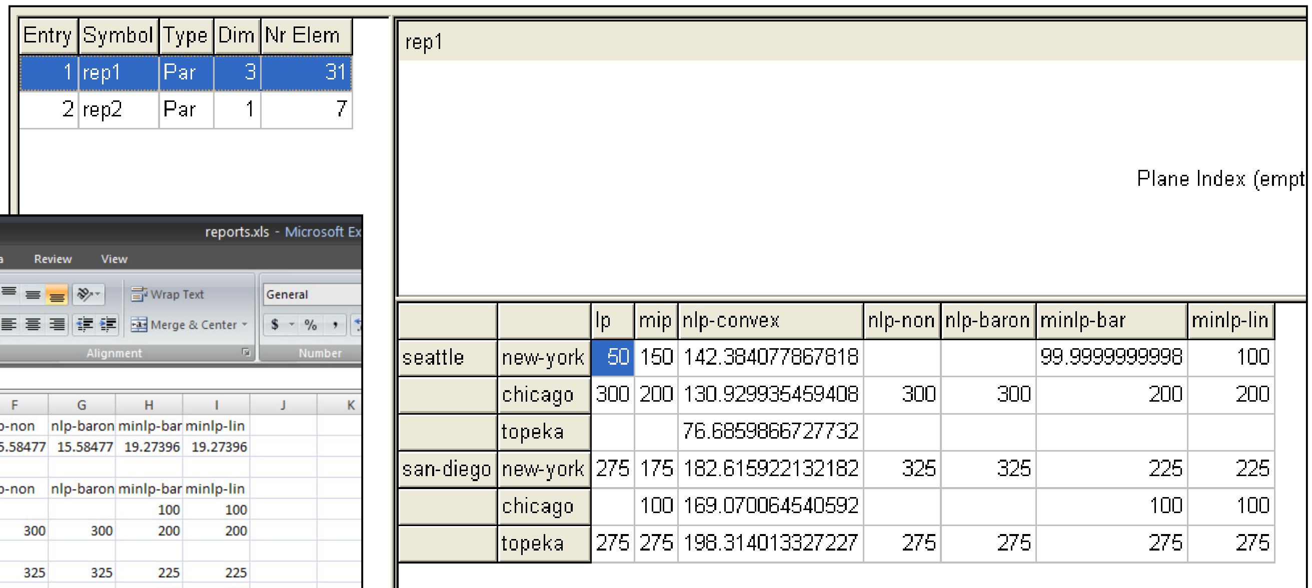
Demo! Put Utility

```
set help(*);
option help<repship;

file fx /results.txt/;

put fx 'Results of different models created on ' system.date /;
put '---------------------------------------------------' / / ;
loop(help,
  put 'Model:' help.te(help) /;
  put '--------------------' / /;
  put 'Objective value:' repcost(help) / /;
  loop((i,j)$repship(i,j,help),
    put 'Shipment from 'i.te(i):10' to 'j.te(j):10' is: 'repship(i,j,help) /;
  );
  put / /;
);
putclose;
```

Demo! GDX and GDXXRW

- execute_unload 'all.gdx';
- gdx=all2

  → gdxdiff

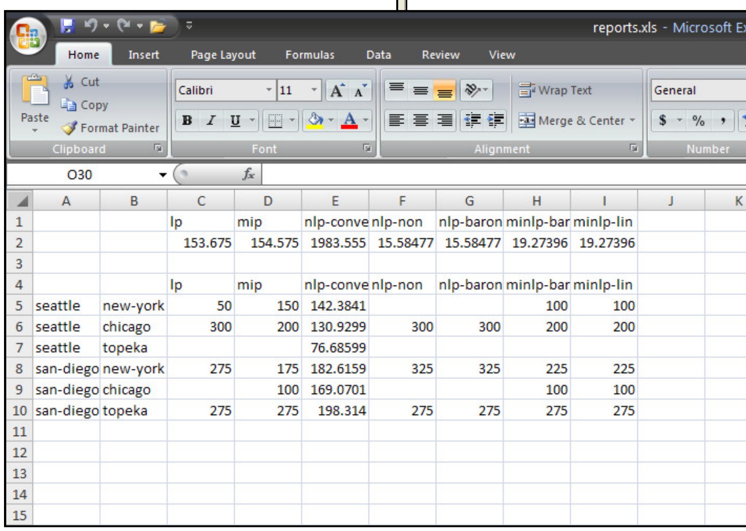


Entry	Symbol	Type	Dim	Nr Elem
1	rep1	Par	3	31
2	rep2	Par	1	7

rep1

Plane Index (empt

		lp	mip	nlp-convex	nlp-non	nlp-baron	minlp-bar	minlp-lin
seattle	new-york	50	150	142.384077867818			99.9999999998	100
	chicago	300	200	130.929935459408	300	300	200	200
	topeka			76.6859866727732				
san-diego	new-york	275	175	182.615922132182	325	325	225	225
	chicago		100	169.070064540592			100	100
	topeka	275	275	198.314013327227	275	275	275	275



	A	B	C	D	E	F	G	H	I
1			lp	mip	nlp-conve	nlp-non	nlp-baron	minlp-bar	minlp-lin
2			153.675	154.575	1983.555	15.58477	15.58477	19.27396	19.27396
3									
4			lp	mip	nlp-conve	nlp-non	nlp-baron	minlp-bar	minlp-lin
5	seattle	new-york	50	150	142.3841			100	100
6	seattle	chicago	300	200	130.9299	300	300	200	200
7	seattle	topeka			76.68599				
8	san-diego	new-york	275	175	182.6159	325	325	225	225
9	san-diego	chicago		100	169.0701			100	100
10	san-diego	topeka	275	275	198.314	275	275	275	275

- execute_unload 'reports.gdx' repcost, repship;
  execute 'gdxxrw reports.gdx par=repcost cdim=1 rdim=0 rng=Report!c1';
  execute 'gdxxrw reports.gdx par=repship cdim=1 rdim=2 rng=Report!a4';

Demo! Capture an Instance

- GAMS “solver”: convert
  - gams mymodel modeltype=convert

  or
  - option minlp=convert;
    solve m4 using minlp minimizing z;

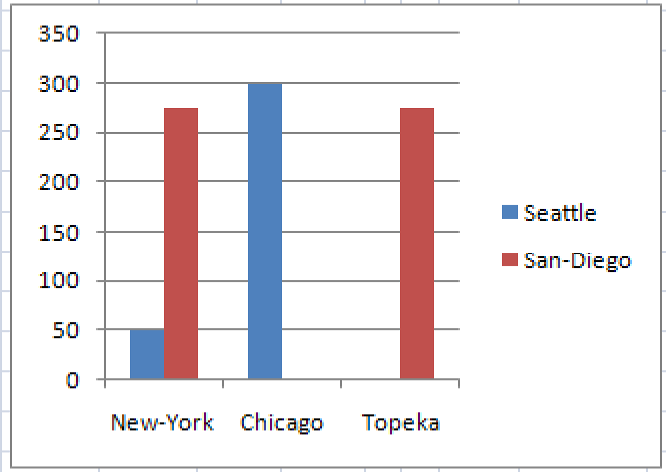
  → anonymized scalar model gams.gms and dictionary dict.txt

  → translation into format required by other tools
    - mps
    - mpi
    - oml
    - …

Demo! Starting a model from a spreadsheet

	A	B	C	D	E	F
1	**Distance**	New-York	Chicago	Topeka		Supply
2	Seattle	2.5	1.7	1.8		350
3	San-Diego	2.5	1.8	1.4		600
4						
5	Demand	325	300	275		
6						
7	Freight cost	90				
8						
9						
10	**SHIPMENT**	New-York	Chicago	Topeka		
11	Seattle	50	300	0		
12	San-Diego	275	0	275		

18	**GAMS Directory:**	c:\program files\GAMS23.2\
19	**Working Directory:**	c:\tmp\
20	**Solver:**	CPLEX



Clear Solution

SOLVE LP

SOLVE MIP

Solver: CPLEX
Equations: 6 Variables: 7
Model Status: 1 Optimal
Solver Status: 1 Normal Completion
Iterations: 4 Solve Time: 0.00

Objective Value: 153.675

Contacting GAMS

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

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
support@gams-software.com