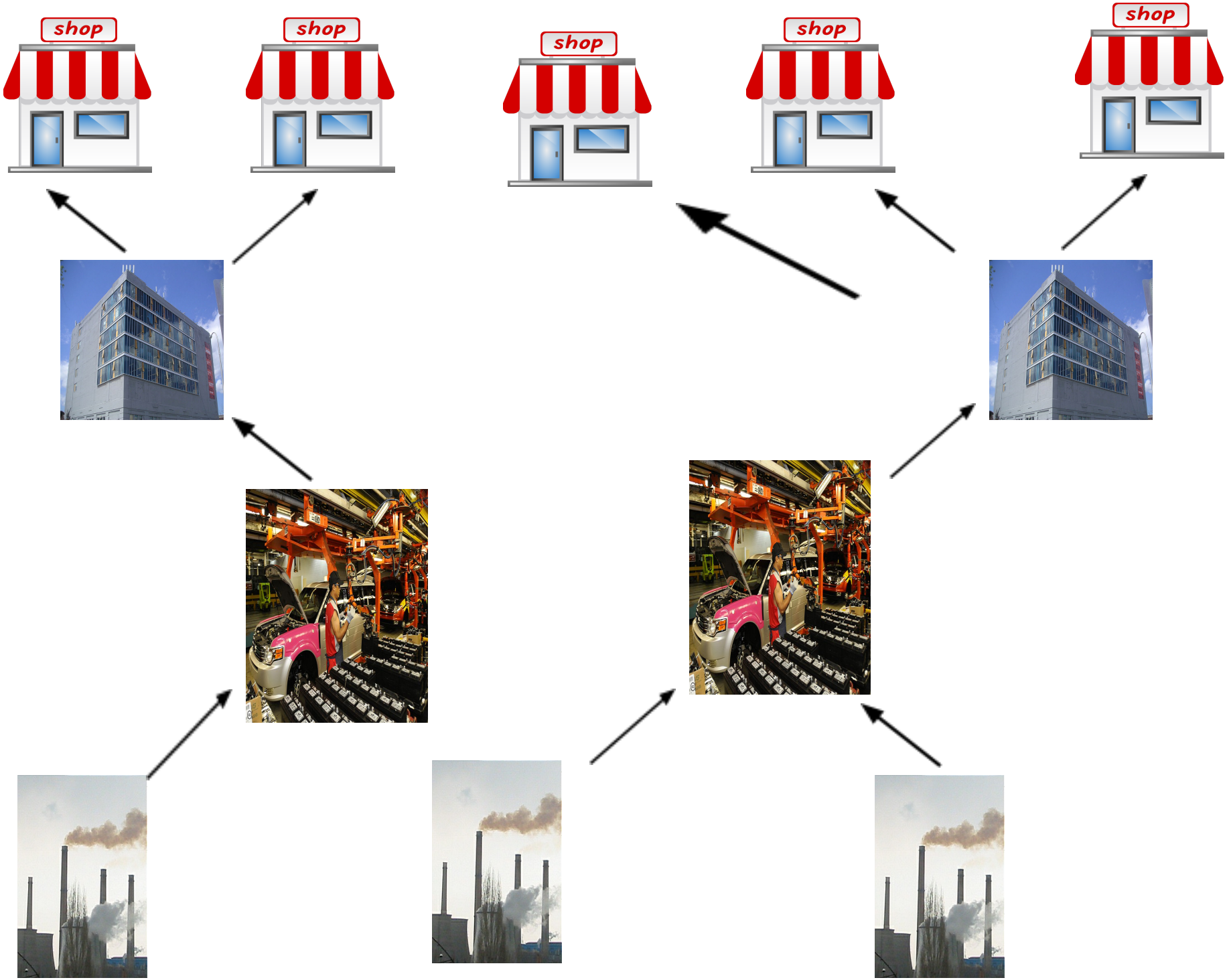
Selecting an Optimiser

for a multinational supply chain - real world case study

Eddy Parkinson

Multi-National Supply Chain
Customers in China/USA/Australia etc


shop
shop
shop
shop
shop

- Exchange Rates
- Import Tax
- Transport Costs
- Factory Costs
- Warehouse Costs
- 12 Month Plan

Meta-Heuristic or Solver ?

Solver

Generation Language

1 GL	Machine code
2 GL	Assembler
3 GL	C++, Java, BASIC
4 GL	SQL, PHP
5 GL	Prolog, Solvers

Order of magnitude improvements

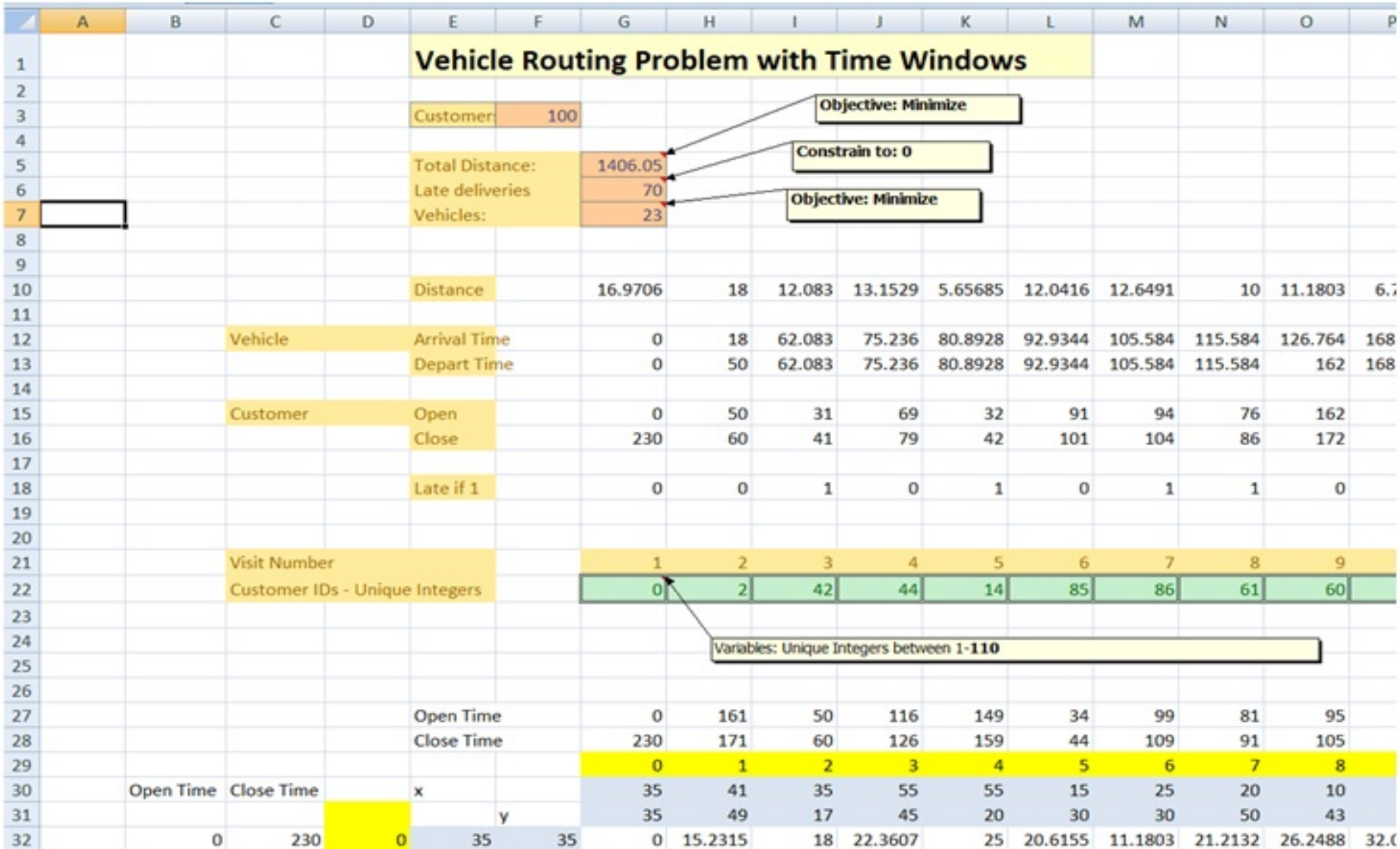
2 GL
Software Libraries (Code reuse)
3 GL
4 GL
5 GL
Developer GUI tools

Meta-heuristics & old innovations

	Meta-heuristics	Solvers
2 GL		
Software Libraries (Code reuse)	✓	
3 GL	✓	
4 GL		
5 GL		✓
Developer GUI tools		✓

2 Issues

VRPTW with a Solver



Vehicle Routing Problem with Time Windows

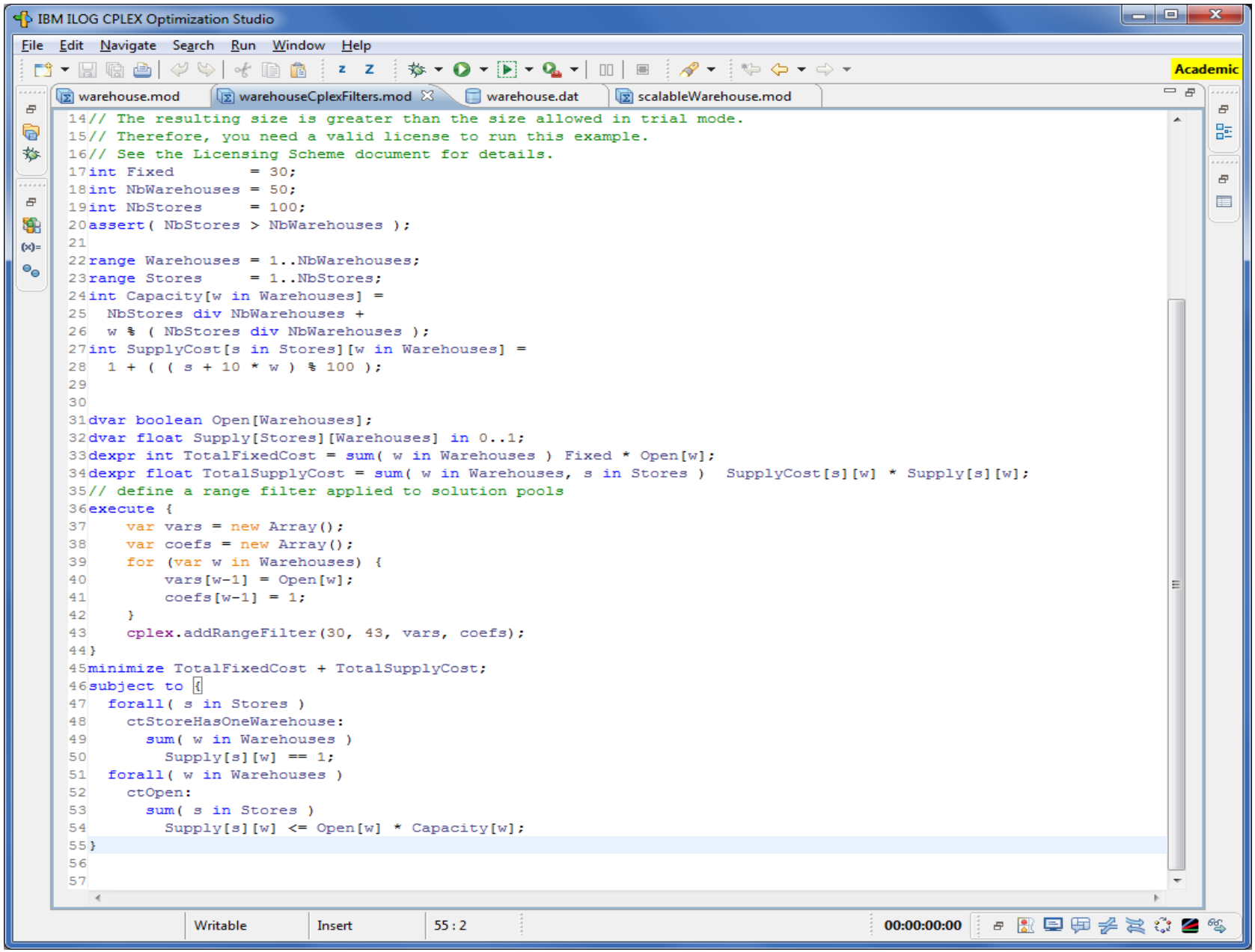
				Customers:	100											
				Total Distance:		1406.05										
				Late deliveries		70										
				Vehicles:		23										
				Distance		16.9706	18	12.083	13.1529	5.65685	12.0416	12.6491	10	11.1803	6.7
		Vehicle		Arrival Time		0	18	62.083	75.236	80.8928	92.9344	105.584	115.584	126.764	168
				Depart Time		0	50	62.083	75.236	80.8928	92.9344	105.584	115.584	162	168
		Customer		Open		0	50	31	69	32	91	94	76	162	
				Close		230	60	41	79	42	101	104	86	172	
				Late if 1		0	0	1	0	1	0	1	1	0	
		Visit Number				1	2	3	4	5	6	7	8	9	
		Customer IDs - Unique Integers				0	2	42	44	14	85	86	61	60	
				Open Time		0	161	50	116	149	34	99	81	95	
				Close Time		230	171	60	126	159	44	109	91	105	
						0	1	2	3	4	5	6	7	8	
	Open Time	Close Time		x		35	41	35	55	55	15	25	20	10	
					y	35	49	17	45	20	30	30	50	43	
	0	230	0	35	35	0	15.2315	18	22.3607	25	20.6155	11.1803	21.2132	26.2488	32.0

Objective: Minimize

Constrain to: 0

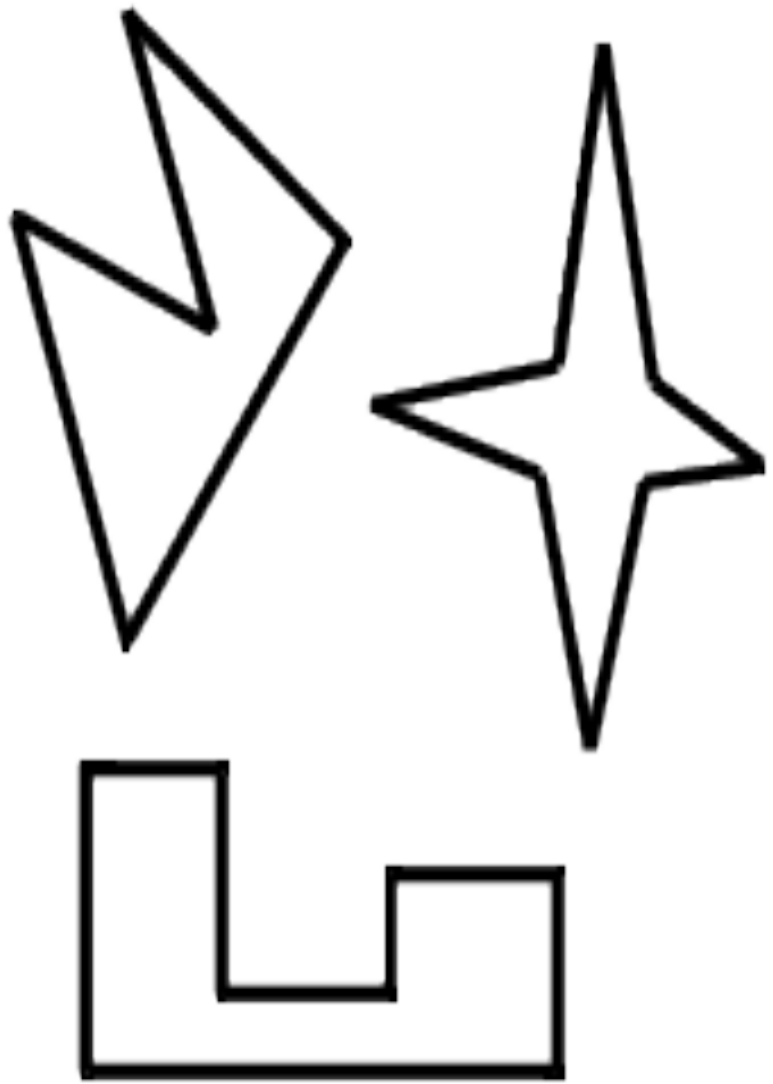
Objective: Minimize

Variables: Unique Integers between 1-**110**

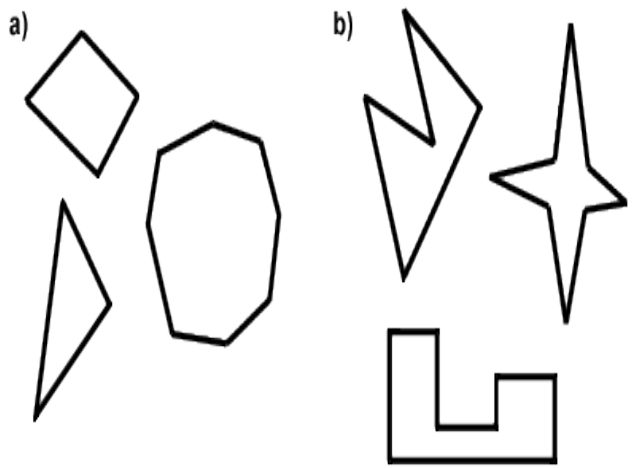

IBM ILOG CPLEX Optimization Studio
File Edit Navigate Search Run Window Help
Academic
warehouse.mod
warehouseCplexFilters.mod
warehouse.dat
scalableWarehouse.mod
// The resulting size is greater than the size allowed in trial mode.
// Therefore, you need a valid license to run this example.
// See the Licensing Scheme document for details.
int Fixed = 30;
int NbWarehouses = 50;
int NbStores = 100;
assert(NbStores > NbWarehouses);
range Warehouses = 1..NbWarehouses;
range Stores = 1..NbStores;
int Capacity[w in Warehouses] =
NbStores div NbWarehouses +
w % (NbStores div NbWarehouses);
int SupplyCost[s in Stores][w in Warehouses] =
1 + ((s + 10 * w) % 100);
dvar boolean Open[Warehouses];
dvar float Supply[Stores][Warehouses] in 0..1;
dexpr int TotalFixedCost = sum(w in Warehouses) Fixed * Open[w];
dexpr float TotalSupplyCost = sum(w in Warehouses, s in Stores) SupplyCost[s][w] * Supply[s][w];
// define a range filter applied to solution pools
execute {
var vars = new Array();
var coefs = new Array();
for (var w in Warehouses) {
vars[w-1] = Open[w];
coefs[w-1] = 1;
}
cplex.addRangeFilter(30, 43, vars, coefs);
}
minimize TotalFixedCost + TotalSupplyCost;
subject to {
forall(s in Stores)
ctStoreHasOneWarehouse:
sum(w in Warehouses)
Supply[s][w] == 1;
forall(w in Warehouses)
ctOpen:
sum(s in Stores)
Supply[s][w] <= Open[w] * Capacity[w];
}
Writable
Insert
55 : 2
00:00:00:00

a)

b)



Meta-Heuristic OR Solver ?



Vehicle Routing Problem with Time Windows

Customers: 100

Total Distance: 1406.05

Late deliveries: 70

Vehicles: 23

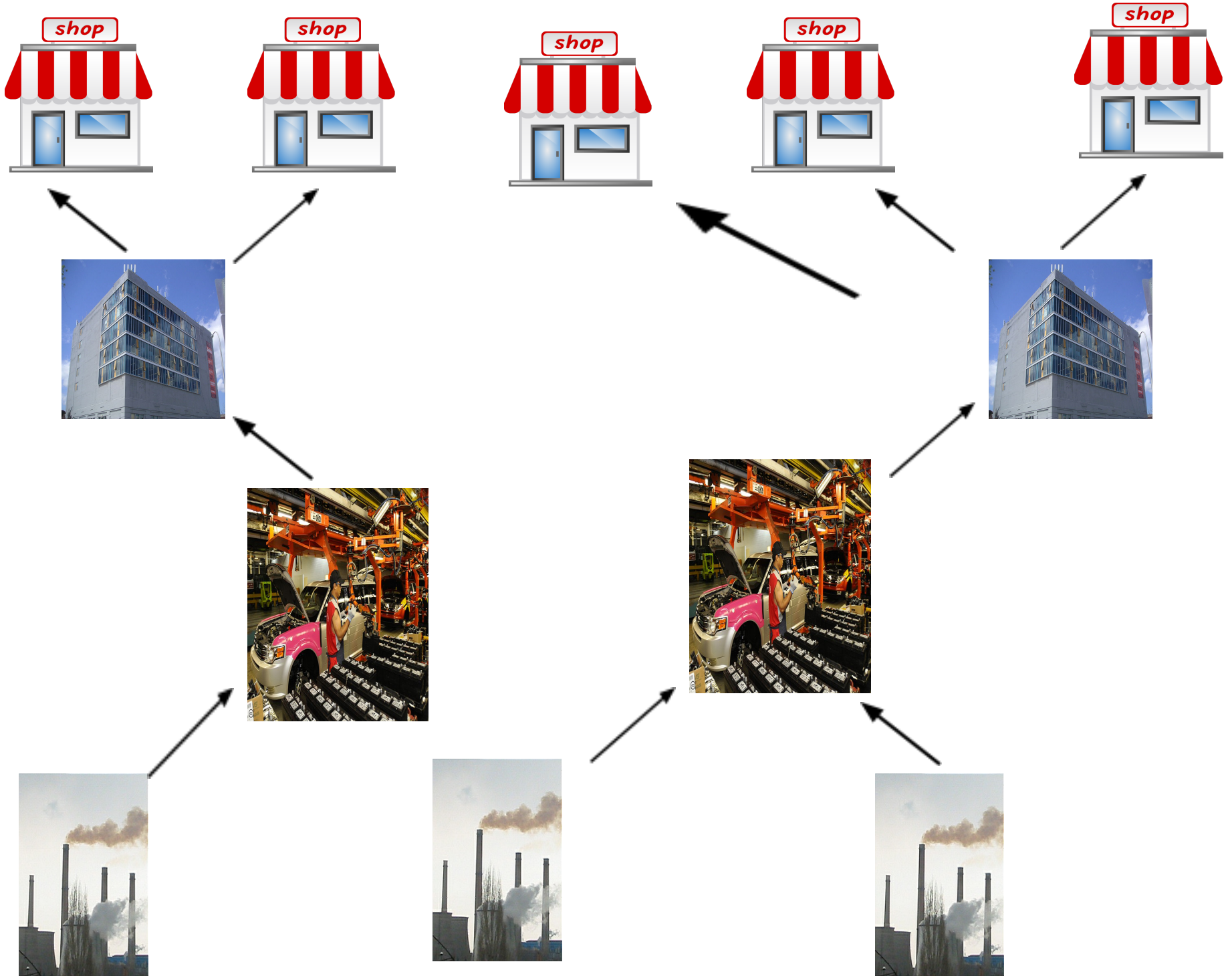
Objective: Minimize

Constrain to: 0

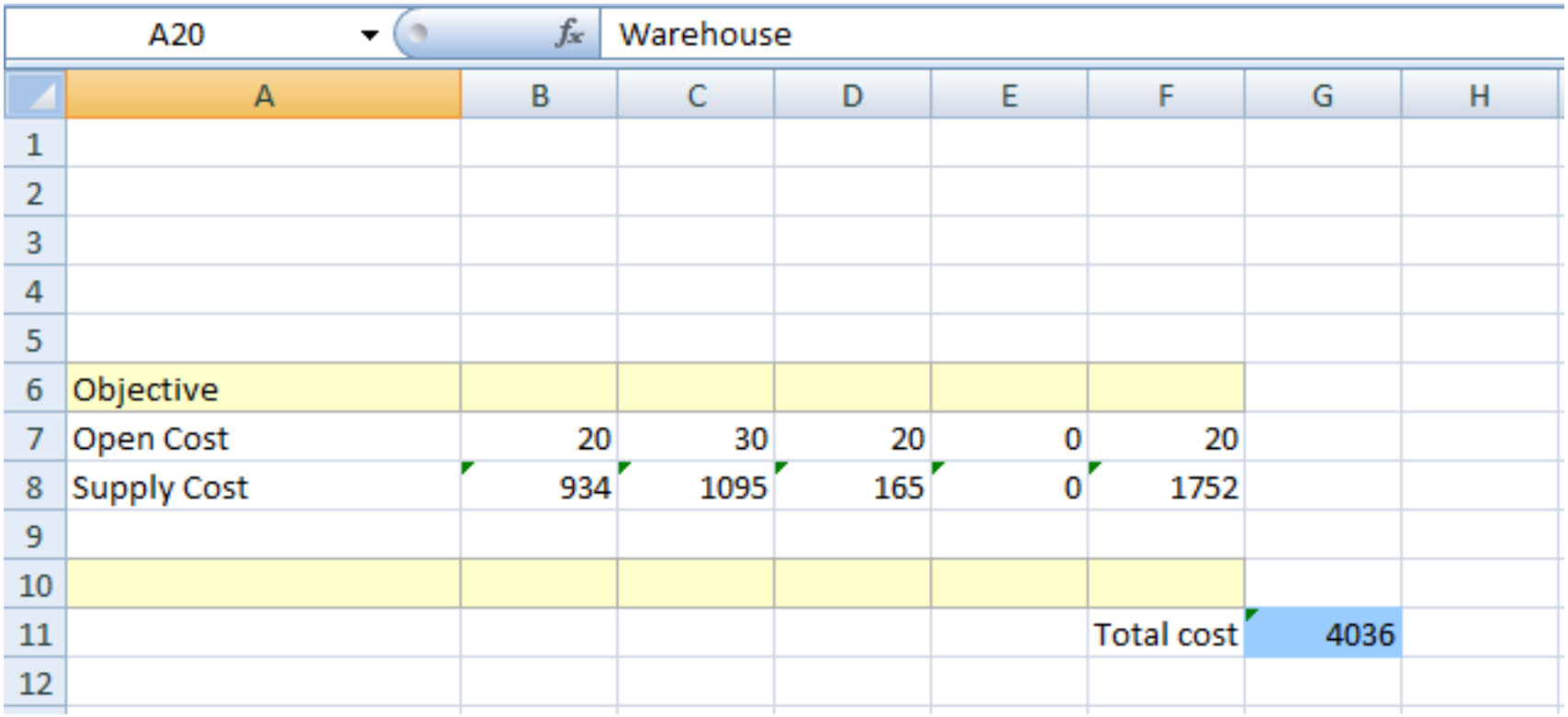
Objective: Minimize

Distance					16.9706	18	12.083	13.1529	5.65685	12.0416	12.6491	10	11.1803	6.7
Vehicle	Arrival Time				0	18	62.083	75.236	80.8928	92.9344	105.584	115.584	126.764	168
	Depart Time				0	50	62.083	75.236	80.8928	92.9344	105.584	115.584	162	168
Customer	Open				0	50	31	69	32	91	94	76	162	
	Close				230	60	41	79	42	101	104	86	172	
	Late if 1				0	0	1	0	1	0	1	1	0	
Visit Number					1	2	3	4	5	6	7	8	9	
Customer IDs - Unique Integers					0	2	42	44	14	85	86	61	60	
	Open Time				0	161	50	116	149	34	99	81	95	
	Close Time				230	171	60	126	159	44	109	91	105	
					0	1	2	3	4	5	6	7	8	
Open Time	Close Time		x		35	41	35	55	55	15	25	20	10	
			y		35	49	17	45	20	30	30	50	43	
0	230	0	35	35	0	15.2315	18	22.3607	25	20.6155	11.1803	21.2132	26.2488	32.0

Variables: Unique Integers between 1-110


shop
shop
shop
shop
shop

Objective Function



A20 | Warehouse

	A	B	C	D	E	F	G	H
1								
2								
3								
4								
5								
6	Objective							
7	Open Cost	20	30	20	0	20		
8	Supply Cost	934	1095	165	0	1752		
9								
10								
11						Total cost	4036	
12								

Raw Data

13										
14										
15	Data									
16	*Warehouse*	*1*	*2*	*3*	*4*	*5*				
17	Open Cost	20	30	20	30	20				
18										
19	Transport cost									
20	*Warehouse*	*1*	*2*	*3*	*4*	*5*			Demand From store	
21	Store 1	20	24	11	25	30		Store 1	15	
22	Store 2	28	27	82	83	74		Store 2	30	
23	Store 3	74	97	71	96	70		Store 3	24	
24	Store 4	2	55	73	69	61		Store 4	24	
25	Store 5	46	96	59	83	4		Store 5	18	
26	Store 6	42	22	29	67	59		Store 6	5	
27	Store 7	1	5	73	59	56		Store 7	12	
28	Store 8	10	73	13	43	96		Store 8	31	
29	Store 9	93	35	63	85	46		Store 9	5	
30	Store 10	47	65	55	71	95		Store 10	12	
31										
32	*Warehouse*	*1*	*2*	*3*	*4*	*5*				
33	Capacity	120	70	90	20	80				
34										

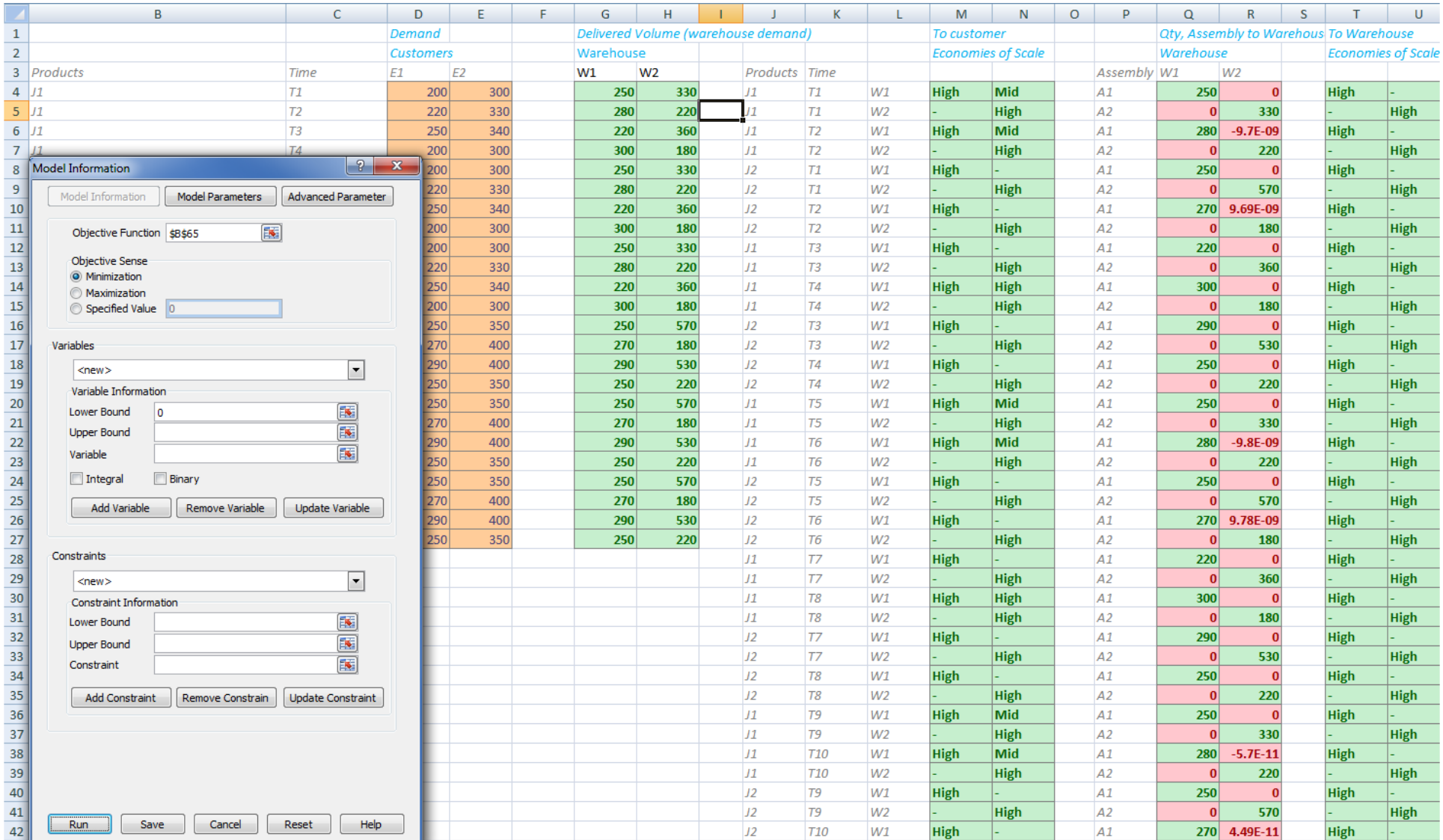
Constraints

72						
73						
74	Constraints					
75	Store has warehouse	Demand		Supply		
76	Store 1	15	=	15		
77	Store 2	30	=	30		
78	Store 3	24	=	24		
79	Store 4	24	=	24		
80	Store 5	18	=	18		
81	Store 6	5	=	5		
82	Store 7	12	=	12		
83	Store 8	31	=	31		
84	Store 9	5	=	5		
85	Store 10	12	=	12		
86						
87	Supplied warehouses cou	79	40	15	0	42
88		<=	<=	<=	=<=	<=
89	Max Capacity	120	70	90	0	80
90						

Variables

A20 | Warehouse

	A	B	C	D	E	F	G	H
52								
53								
54	Variables							
55	*Warehouse*	*1*	*2*	*3*	*4*	*5*		
56	Open / Closed	**1**	**1**	**1**	**0**	**1**		
57								
58								
59		Qty to transport						
60	*Warehouse*	*1*	*2*	*3*	*4*	*5*		
61	Store 1	**0**	**0**	**15**	**0**	**0**		
62	Store 2	**0**	**30**	**0**	**0**	**0**		
63	Store 3	**0**	**0**	**0**	**0**	**24**		
64	Store 4	**24**	**0**	**0**	**0**	**0**		
65	Store 5	**0**	**0**	**0**	**0**	**18**		
66	Store 6	**0**	**5**	**0**	**0**	**0**		
67	Store 7	**12**	**0**	**0**	**0**	**0**		
68	Store 8	**31**	**0**	**0**	**0**	**0**		
69	Store 9	**0**	**5**	**0**	**0**	**0**		
70	Store 10	**12**	**0**	**0**	**0**	**0**		
71								
72								



	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1			Demand			Delivered Volume (warehouse demand)						To customer				Qty, Assembly to Warehous			To Warehouse	
2			Customers			Warehouse						Economies of Scale				Warehouse			Economies of Scale	
3	Products	Time	E1	E2		W1	W2		Products	Time					Assembly	W1	W2			
4	J1	T1	200	300		250	330		J1	T1	W1	High	Mid		A1	250	0		High	-
5	J1	T2	220	330		280	220		J1	T1	W2	-	High		A2	0	330		-	High
6	J1	T3	250	340		220	360		J1	T2	W1	High	Mid		A1	280	-9.7E-09		High	-
7	J1	T4	200	300		300	180		J1	T2	W2	-	High		A2	0	220		-	High
8			200	300		250	330		J2	T1	W1	High	-		A1	250	0		High	-
9			220	330		280	220		J2	T1	W2	-	High		A2	0	570		-	High
10			250	340		220	360		J2	T2	W1	High	-		A1	270	9.69E-09		High	-
11			200	300		300	180		J2	T2	W2	-	High		A2	0	180		-	High
12			200	300		250	330		J1	T3	W1	High	-		A1	220	0		High	-
13			220	330		280	220		J1	T3	W2	-	High		A2	0	360		-	High
14			250	340		220	360		J1	T4	W1	High	High		A1	300	0		High	-
15			200	300		300	180		J1	T4	W2	-	High		A2	0	180		-	High
16			250	350		250	570		J2	T3	W1	High	-		A1	290	0		High	-
17			270	400		270	180		J2	T3	W2	-	High		A2	0	530		-	High
18			290	400		290	530		J2	T4	W1	High	-		A1	250	0		High	-
19			250	350		250	220		J2	T4	W2	-	High		A2	0	220		-	High
20			250	350		250	570		J1	T5	W1	High	Mid		A1	250	0		High	-
21			270	400		270	180		J1	T5	W2	-	High		A2	0	330		-	High
22			290	400		290	530		J1	T6	W1	High	Mid		A1	280	-9.8E-09		High	-
23			250	350		250	220		J1	T6	W2	-	High		A2	0	220		-	High
24			250	350		250	570		J2	T5	W1	High	-		A1	250	0		High	-
25			270	400		270	180		J2	T5	W2	-	High		A2	0	570		-	High
26			290	400		290	530		J2	T6	W1	High	-		A1	270	9.78E-09		High	-
27			250	350		250	220		J2	T6	W2	-	High		A2	0	180		-	High
28									J1	T7	W1	High	-		A1	220	0		High	-
29									J1	T7	W2	-	High		A2	0	360		-	High
30									J1	T8	W1	High	High		A1	300	0		High	-
31									J1	T8	W2	-	High		A2	0	180		-	High
32									J2	T7	W1	High	-		A1	290	0		High	-
33									J2	T7	W2	-	High		A2	0	530		-	High
34									J2	T8	W1	High	-		A1	250	0		High	-
35									J2	T8	W2	-	High		A2	0	220		-	High
36									J1	T9	W1	High	Mid		A1	250	0		High	-
37									J1	T9	W2	-	High		A2	0	330		-	High
38									J1	T10	W1	High	Mid		A1	280	-5.7E-11		High	-
39									J1	T10	W2	-	High		A2	0	220		-	High
40									J2	T9	W1	High	-		A1	250	0		High	-
41									J2	T9	W2	-	High		A2	0	570		-	High
42									J2	T10	W1	High	-		A1	270	4.49E-11		High	-

Google:Spreadsheet Analytics

- ***Premium Solver, Premium Solver Platform, RiskSolver Platform*** *by Frontline Systems*
- ***Evolver*** *by Palisade Corporation*
- ***Generator*** *by New Light Industries*
- ***OptWorks*** *by SpaceWorks Software*
- ***Solve XL*** *by Diapason Consulting*
- ***GANetXL*** *by University of Exeter*
- ***GeneHunter*** *by Ward Systems*
-

Questions ?

- Exchange Rates
- Import Tax
- Transport Costs
- Factory Costs
- Warehouse Costs
- 12 Month Plan