International Atomic Energy Agency

An open source energy planning approach: SOFT-MESSAGE

June 2008
Paris, France

M. Howells, H.H. Rogner, I. Jalal & M. Isshiki
Planning and Economic Studies Section

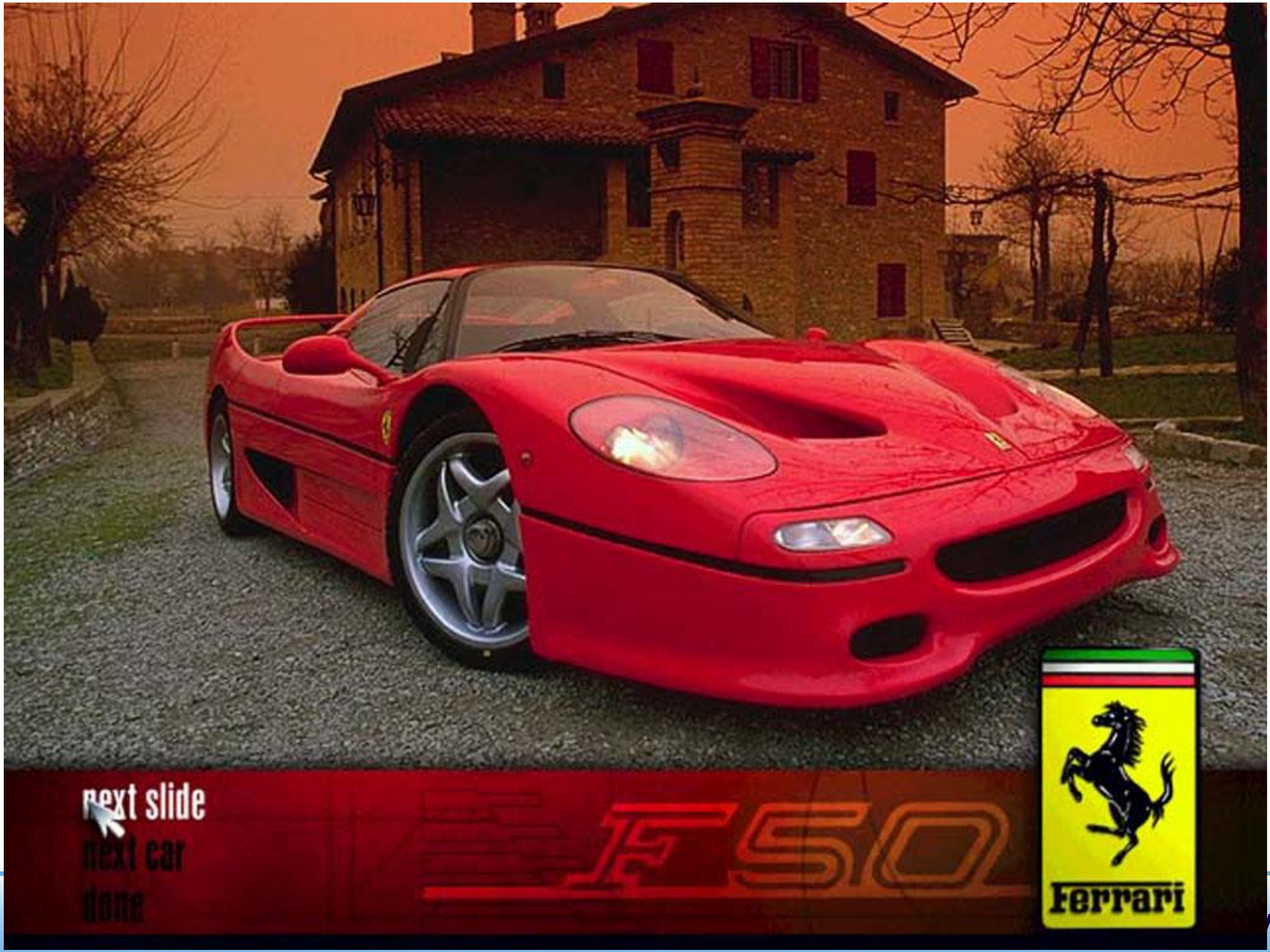
Outline

- Objective: Component driven free evolvable optimization model
- View of some of the components
  - Interface, envelope, results viewer, model code etc.
- Basic equations
- Simple application
- Conclusions

Selected objectives

- **S**imple **O**pen **F**lexible **T**ransparent
- MESSAGE (like MARKAL, TIMES, EFOM etc) is part of the Hefele-Manne family of models (limited but powerful)
- Reduce the barriers for uptake and basic capacity in the use of optimization models
  - For application (limited set of national medium term modeling)
  - For development (very easy access to all equations)
- Free and unrestricted distribution (all components)
- Should increase the need and use of more powerful approaches
- Improve the power of simple but popular approaches
- As development is open, evolution can be sustainable
- (Target is a limited set of medium term applications)

What we would like to see

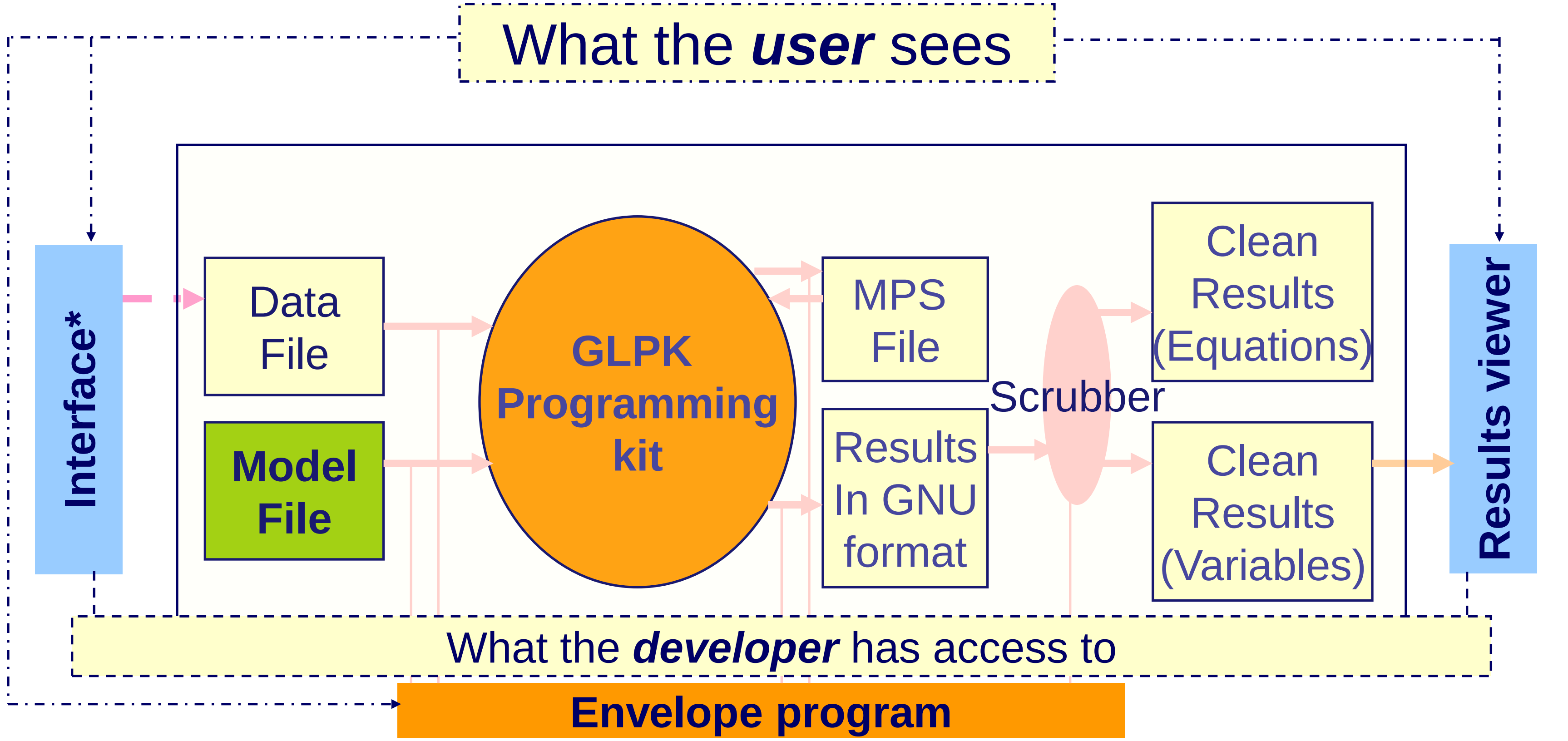


What I am going to show you..

Component driven

- Components that do not “lock us in”
- All aspects are “open” many service providers / programmers / enthusiasts
- Components are (as) independent (as possible)
- Can be changed, improved, replaced
- Components so far….

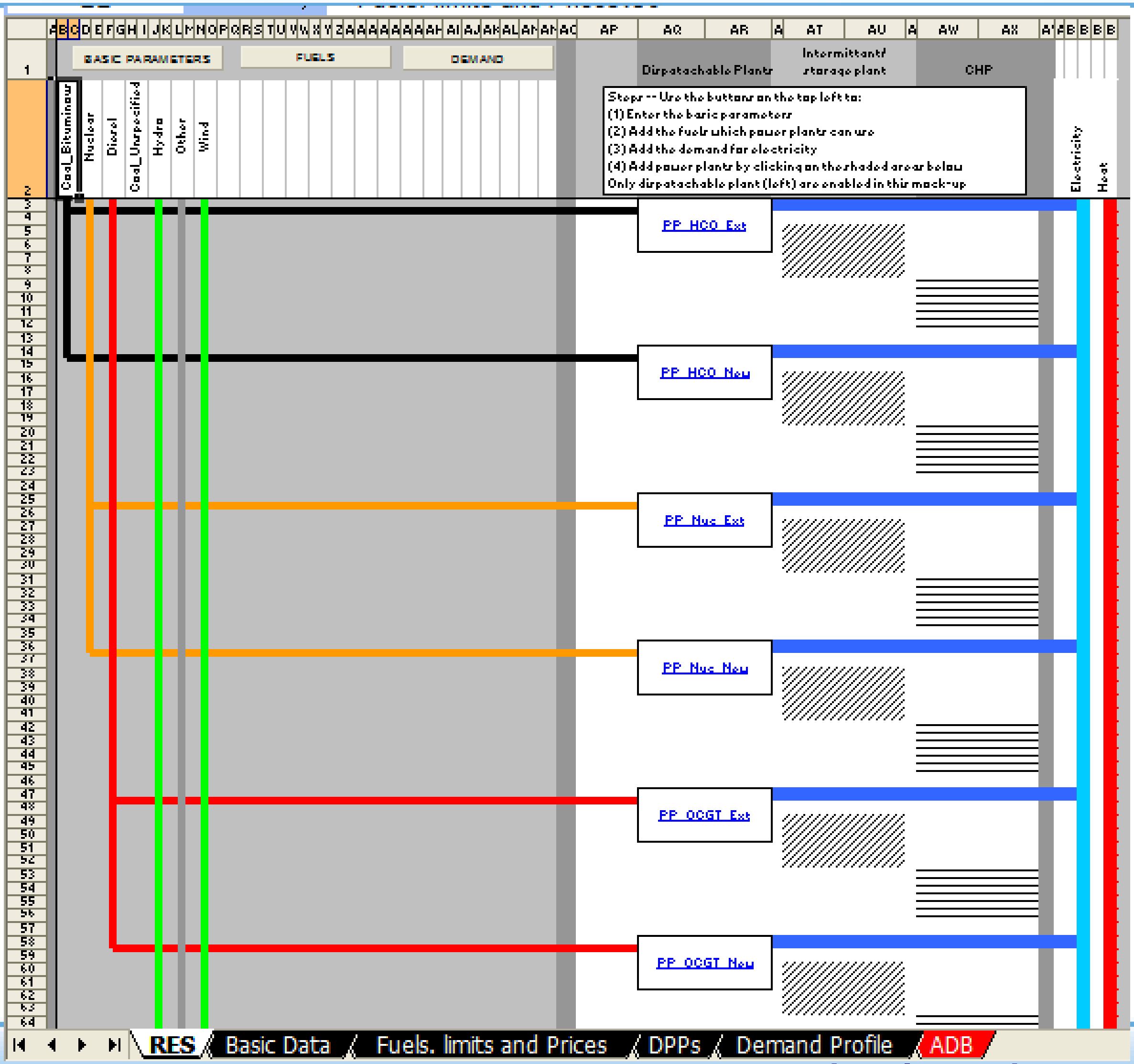
Components



* Interface is ONLY for testing. It is also independent – could use LEAP – could use more symbolic interface – RES type interface.

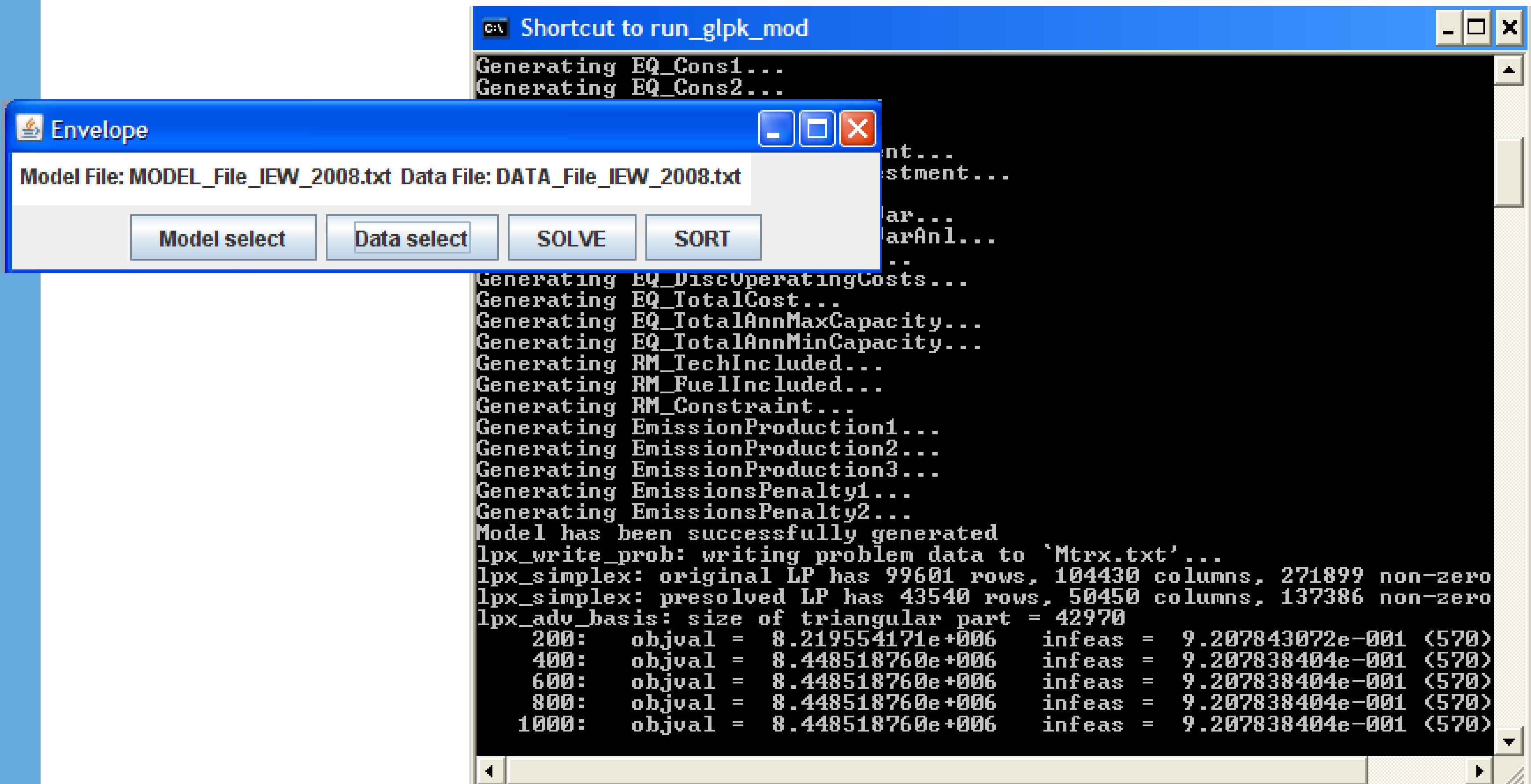
Functional Mock-up interface

(To be scrapped soon. It was compiled to help testing. Again, the move is to make this as independent from specific implementation model implementation. We can also focus on simple intuitive interfaces. Could theoretically use the same model with a different interface: LEAP / ANSWER / VEDA etc)



Envelope Program

(Written in Java this picks up the Model file and Data file and then submits the run. The data is then cleaned for viewing / analysis)



Results viewer

(Written in Java. Excel breaks down when results sets are large. Again, this utility is completely independent of the interface, model file, data file etc.)

Display result data file

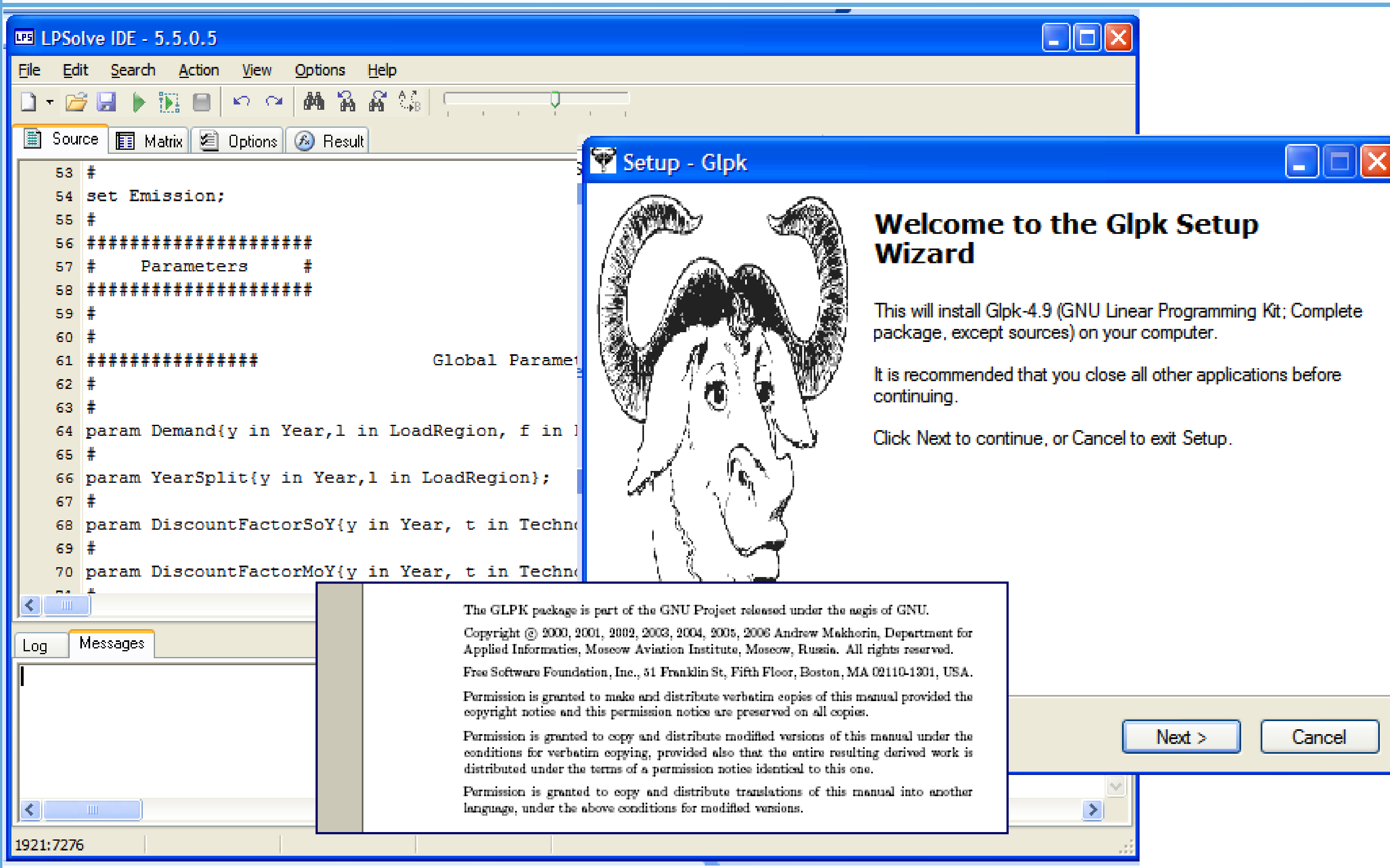
Data select

Click on the Data File button to choose the data to be shown on the table.
Click on the table headers to activate sorting

ID	Varible...	Year	ID2	ID3	ID4	St	Activity	Lowe...	Marginal	eps
	TotCapAr			PP_HCO_Ext						
19209	TotCap...	My2005	none	PP_HCO_Ext	none	B	35218	0	n/a	n/a
19229	TotCap...	My2006	none	PP_HCO_Ext	none	B	35218	0	n/a	n/a
19249	TotCap...	My2007	none	PP_HCO_Ext	none	B	35218	0	n/a	n/a
19269	TotCap...	My2008	none	PP_HCO_Ext	none	B	35218	0	n/a	n/a
19289	TotCap...	My2009	none	PP_HCO_Ext	none	B	35218	0	n/a	n/a
19309	TotCap...	My2010	none	PP_HCO_Ext	none	B	35218	0	n/a	n/a
19329	TotCap...	My2011	none	PP_HCO_Ext	none	B	35218	0	n/a	n/a
19349	TotCap...	My2012	none	PP_HCO_Ext	none	B	35218	0	n/a	n/a
19369	TotCap...	My2013	none	PP_HCO_Ext	none	B	35218	0	n/a	n/a
19389	TotCap...	My2014	none	PP_HCO_Ext	none	B	35218	0	n/a	n/a
19409	TotCap...	My2015	none	PP_HCO_Ext	none	B	35218	0	n/a	n/a
19429	TotCap...	My2016	none	PP_HCO_Ext	none	B	34543	0	n/a	n/a
19449	TotCap...	My2017	none	PP_HCO_Ext	none	B	34543	0	n/a	n/a
19469	TotCap...	My2018	none	PP_HCO_Ext	none	B	34543	0	n/a	n/a
19489	TotCap...	My2019	none	PP_HCO_Ext	none	B	34543	0	n/a	n/a
19509	TotCap...	My2020	none	PP_HCO_Ext	none	B	34543	0	n/a	n/a
19529	TotCap...	My2021	none	PP_HCO_Ext	none	B	31868	0	n/a	n/a
19549	TotCap...	My2022	none	PP_HCO_Ext	none	B	29768	0	n/a	n/a
19569	TotCap...	My2023	none	PP_HCO_Ext	none	B	29768	0	n/a	n/a
19589	TotCap...	My2024	none	PP_HCO_Ext	none	B	29768	0	n/a	n/a
19609	TotCap...	My2025	none	PP_HCO_Ext	none	B	29768	0	n/a	n/a
19629	TotCap...	My2026	none	PP_HCO_Ext	none	B	29768	0	n/a	n/a

GLPK Programming Kit

(A freeware subset of the AMPL mathematical programming language. Pictured is a development kit (IDE Solve by Henri Gourvest), part of the manual and the Glpk Setup)



```
# Engine for SOFT-MESSAGE pre-mockup 1b
# By Mark Idwal Howells
#
##############################################
#                                                    #
#Satisfying Capacity, demand & consumption           #
#          1.Including: multiple technologies        #
#          2.Including: multiple energy demands      #
#          3.Including: Operating & Capital Costs    #
#          4.Including: Load Regions                 #
#          5.Including: Multiple fuels               #
#          6.Including: Avlblty & Cpcty Factor       #
#          7.Including: Salvage Cost                 #
#          *8. Including: Storage (commented out)    #
#          9. Including: Emissions                   #
#          10. Including: 2 Activities per tec       #
#          11. Including: DF for SoY and MoY         #
#                                                    #
##############################################
#
#
#
#                        ######################################
#######################          Model Definition          ############
#                        ######################################
#
#
#
####################
#   Model Sets      #
####################
#
#
set Year;
set Vintage;
set Technology;
set LoadRegion;
set Fuel;
set BoundryInstances;
set Storage;
set River;
set Emission;
#
####################
#   Parameters      #
####################
#
#
###############          Global Parameters          ############
#
#
param Demand{y in Year,l in LoadRegion, f in Fuel};
param YearSplit{y in Year,l in LoadRegion};
param DiscountFactorSoY{y in Year, t in Technology};
param DiscountFactorMoY{y in Year, t in Technology};
param IDMatrix{y in Year, v in Vintage};
param OneMatrixYear{y in Year};
param OneMatrixVintage{v in Vintage};
param OneMatrixLR{l in LoadRegion};
param OneMatrixFuel{f in Fuel};
#param OneMatrixTechnology{t in Technology};
param OneMatrixBoundryInstances{b in BoundryInstances};
#param OneMatrixStorage{s in Storage};
#
################          Technology Parameters          ############
#
#
param CapacityFactor{y in Year, t in Technology};
param AvailabilityFactor{y in Year, t in Technology};
param VintageMatrix{y in Year, v in Vintage, t in Technology};
param SalvageFactor{y in Year, t in Technology};
param ResidualCapacity{y in Year, t in Technology};
param OtptActvtyRatio1{y in Year, t in Technology, f in Fuel};
param OtptActvtyRatio2{y in Year, t in Technology, f in Fuel};
param InptActvtyRatio1{y in Year, t in Technology, f in Fuel};
param InptActvtyRatio2{y in Year, t in Technology, f in Fuel};
#
#
#########     Technology Costs                              ############
#
param CapitalCost{y in Year, t in Technology};
param VariableCost1{y in Year, t in Technology};
param VariableCost2{y in Year, t in Technology};
param FixedCost{y in Year, t in Technology};
#
#########                Storage Parameters                 ############
#
#param StorageConstraint{y in Year, l in LoadRegion, b in BoundryInstances};
#param StartLevel{s in Storage};
#param RiverInflow{r in River, y in Year, l in LoadRegion};
#param RiverOutflow{r in River, y in Year, l in LoadRegion};
#param RiverToStorage{r in River, s in Storage};
#param FlowDump{r in River, s in Storage};
#param TechnologyToStorage1{t in Technology, s in Storage};
#
#
#########                Capacity Constraints               ############
#
param TotalAnnMaxCapacity{y in Year, t in Technology};
param TotalAnnMinCapacity{y in Year, t in Technology};
#
#
#########                Reserve Margin                     ############
#
param RMTagTech{y in Year,t in Technology};
param RMTagFuel{y in Year,f in Fuel};
param RM{y in Year};
#
#########     Emissions & Penalties                         ############
#
#
param EmissionActvtyRatio{y in Year, t in Technology, e in Emission};
```

```
param EmissionsPenalty{y in Year, e in Emission};
#
#
########################
#   Model Variables          #
########################
#
#################                         Capacity Variables                         ##############
#
var NewCap{y in Year, t in Technology} >= 0;
var NewCapVDim{v in Vintage, t in Technology};
var NewCapStructure{y in Year, v in Vintage, t in Technology} >= 0;
var TotCapAnn{y in Year, t in Technology} >= 0;
#
#################                         Activity Variables                         ##############
#
var Activity1{y in Year, l in LoadRegion, t in Technology} >= 0;
var Activity2{y in Year, l in LoadRegion, t in Technology} >= 0;
var TotalActivity{y in Year, l in LoadRegion, t in Technology};
var Production1{y in Year, l in LoadRegion, f in Fuel};
var Production2{y in Year, l in LoadRegion, f in Fuel};
var TotalProduction{y in Year, l in LoadRegion, f in Fuel};
var Consumption1{y in Year, l in LoadRegion, f in Fuel};
var Consumption2{y in Year, l in LoadRegion, f in Fuel};
var TotalConsumption{y in Year, l in LoadRegion, f in Fuel};
#
#################                         Costing Variables                         ##############
#
var CapitalInvestment{y in Year, t in Technology};
var DiscCapitalInvestment{y in Year, t in Technology};
var SalvageValue{y in Year, t in Technology};
var OperatingCost{y in Year, t in Technology};
var DiscOperatingCost{y in Year, t in Technology};
var AnnVarOpCost{y in Year,t in Technology};
var VarOpCost{y in Year, l in LoadRegion, t in Technology};
var TotalDiscCost{y in Year, t in Technology};
#
#
#################                         Storage Variables                         ##############
#
# ***********************************************************************
#var NetStorageCharge{s in Storage, y in Year, l in LoadRegion};
#var StorageLevel{s in Storage, b in BoundryInstances};
#var StorageCharge{s in Storage, y in Year, l in LoadRegion};
#var StorageDischarge{s in Storage, y in Year, l in LoadRegion};
#var StorageOverflow{s in Storage, y in Year, l in LoadRegion} >= 0;
#var RiverFlowAddedD{r in River,y in Year,l in LoadRegion};
#var RiverFlowAddedO{r in River,y in Year,l in LoadRegion};
#var RiverFlowAddedT{r in River,y in Year,l in LoadRegion};
#var RiverFlow{r in River, y in Year, l in LoadRegion};
#
# ***********************************************************************
#
#########                         Reserve Margin                         ##############
#
var TotCapinRM{y in Year};
var DMDinRM{y in Year,l in LoadRegion};
#
#
#########                         Emissions                         ##############
#
var AnnualTechEmission{y in Year, t in Technology, e in Emission};
var DiscAnnTechEmPen{y in Year, t in Technology, e in Emission};
var AnnualTechEmPenByEm{y in Year, t in Technology, e in Emission};
var AnnualTechEmPen{y in Year, t in Technology};
var AnnualEmissions{y in Year, e in Emission};
var EmissionsProd1{y in Year, t in Technology, e in Emission};
var EmissionsProd2{y in Year, t in Technology, e in Emission};
#
########################
# Objective Function         #
########################
#
minimize cost: sum{y in Year, t in Technology} TotalDiscCost[y,t];
#
#######################
# Constraints                 #
#######################
#
# ***********************************************************************
#
#s.t. EQ_RiverFlowAddedD{r in River, y in Year, l in LoadRegion}: sum{s in Storage} FlowDump [r,s] *
StorageDischarge[s,y,l] = RiverFlowAddedD [r,y,l];
#s.t. EQ_RiverFlowAddedO{r in River, y in Year, l in LoadRegion}: sum{s in Storage} FlowDump [r,s] *
StorageOverflow[s,y,l] = RiverFlowAddedO [r,y,l];
#s.t. EQ_RiverFlowAddedT{r in River, y in Year, l in LoadRegion}: RiverFlowAddedD[r,y,l] +
RiverFlowAddedO [r,y,l] = RiverFlowAddedT [r,y,l];
#s.t. EQ_RiverFlow{r in River, y in Year, l in LoadRegion}: RiverInflow[r,y,l] - RiverOutflow[r,y,l] +
RiverFlowAddedT[r,y,l] = RiverFlow[r,y,l];
#s.t. EQ_StorageCharge{s in Storage, y in Year, l in LoadRegion}: sum{r in River} RiverFlow[r,y,l] *
RiverToStorage[r,s] = StorageCharge[s,y,l];
#s.t. EQ_NetStorageCharge{s in Storage, y in Year, l in LoadRegion}: NetStorageCharge[s,y,l] =
StorageCharge[s,y,l] - StorageOverflow[s,y,l] - StorageDischarge[s,y,l];
#s.t. EQ_St1{b in BoundryInstances,s in Storage}: sum{l in LoadRegion, y in Year}
NetStorageCharge[s,y,l]*StorageConstraint[y,l,b] = - StartLevel[s]*OneMatrixBoundryInstances[b] +
StorageLevel[s,b];
#s.t. EQ_St2{b in BoundryInstances, s in Storage}: StorageLevel[s,b] >=20;
#s.t. EQ_St3{b in BoundryInstances, s in Storage}: StorageLevel[s,b] <=500;
#s.t. EQ_StorageDischarge{s in Storage, y in Year, l in LoadRegion}: sum{t in Technology}
Activity1[y,l,t] * TechnologyToStorage1[t,s] * YearSplit[y,l] = StorageDischarge[s,y,l];
#
#
#
#
# ***********************************************************************
#
#
# EQ_RiverFlow{y in Year, l in LoadRegion, r in River}: Inflow[y,l,r] - Outflow[y,l,r] + Dump[y,l,r];
# EQ_Dump{y in Year, l in LoadRegion, r in River}
#
#
#
```

```
################        Capacity sufficient for Activity                ############
#
#
s.t. EQ_NewCapDimChange{v in Vintage, t in Technology}: sum{y in Year}
NewCap[y,t]*IDMatrix[y,v]=NewCapVDim[v,t];
s.t. EQ_Structure_Of_New_Investment{y in Year, v in Vintage, t in Technology}:  NewCapVDim[v,t]*
VintageMatrix[y,v,t] = NewCapStructure[y,v,t];
s.t. EQ_TotCapAnn{y in Year, t in Technology}: sum{v in Vintage} NewCapStructure[y,v,t] =
TotCapAnn[y,t] - ResidualCapacity[y,t];
s.t. Constraint_Capacity{y in Year, l in LoadRegion, t in Technology}: TotCapAnn[y,t] *
CapacityFactor[y,t]* OneMatrixLR[l] >= Activity1[y,l,t] + Activity2[y,l,t];
#
################        Activity includes "planned maintenance"         ############
#
#
s.t. Tot_Activity{y in Year, t in Technology, l in LoadRegion}: Activity1[y,l,t] + Activity2[y,l,t] =
TotalActivity[y,l,t];
s.t. PlannedMaintenance{y in Year, t in Technology}: sum{l in LoadRegion}
TotalActivity[y,l,t]*YearSplit[y,l] <= TotCapAnn[y,t]*CapacityFactor[y,t]* AvailabilityFactor[y,t];
#
################        Production>=Demand+Consumption                  ############
#
#
s.t. EQ_Prod1{y in Year, l in LoadRegion, f in Fuel}: sum{t in Technology}
Activity1[y,l,t]*OtptActvtyRatio1[y,t,f] * OneMatrixLR[l] = Production1[y,l,f];
s.t. EQ_Prod2{y in Year, l in LoadRegion, f in Fuel}: sum{t in Technology}
Activity2[y,l,t]*OtptActvtyRatio2[y,t,f] * OneMatrixLR[l] = Production2[y,l,f];
s.t. EQ_TotProd{y in Year, l in LoadRegion, f in Fuel}: Production1[y,l,f] + Production2[y,l,f] =
TotalProduction[y,l,f];
s.t. EQ_Cons1{y in Year, l in LoadRegion, f in Fuel}: sum{t in Technology}
Activity1[y,l,t]*InptActvtyRatio1[y,t,f] * OneMatrixLR[l] = Consumption1[y,l,f];
s.t. EQ_Cons2{y in Year, l in LoadRegion, f in Fuel}: sum{t in Technology}
Activity2[y,l,t]*InptActvtyRatio2[y,t,f] * OneMatrixLR[l] = Consumption2[y,l,f];
s.t. EQ_TotCons{y in Year, l in LoadRegion, f in Fuel}: Consumption1[y,l,f] + Consumption2[y,l,f] =
TotalConsumption[y,l,f];
s.t. EnBalance{y in Year,l in LoadRegion, f in Fuel}: TotalProduction[y,l,f] >= Demand[y,l,f] +
TotalConsumption[y,l,f];
#
#
################                Captial Investment                      ############
#
#
s.t. EQ_CapitalInvestment{y in Year, t in Technology}: CapitalCost[y,t] * NewCap[y,t] =
CapitalInvestment[y,t];
s.t. EQ_DiscCapitalInvestment{y in Year, t in Technology}:
CapitalInvestment[y,t]*DiscountFactorSoY[y,t] = DiscCapitalInvestment[y,t];
#
#
################                Salvage Costs                           ############
#
#
s.t. EQ_SalvageValue{y in Year, t in Technology}: CapitalCost[y,t] * NewCap[y,t] * SalvageFactor[y,t] =
SalvageValue[y,t];
#
#
################                Operating Costs                         ############
#
#
s.t. EQ_OperatingCostsVar{y in Year,l in LoadRegion, t in Technology}:
Activity1[y,l,t]*VariableCost1[y,t]*YearSplit[y,l] + Activity2[y,l,t]*VariableCost2[y,t]*YearSplit[y,l]=
VarOpCost[y,l,t];
s.t. EQ_OperatingCostsVarAnl{y in Year,t in Technology}: sum {l in LoadRegion} VarOpCost[y,l,t] =
AnnVarOpCost[y,t];
s.t. EQ_OperatingCosts{y in Year,t in Technology}: TotCapAnn[y,t]*FixedCost[y,t]+AnnVarOpCost[y,t]
= OperatingCost[y,t];
s.t. EQ_DiscOperatingCosts{y in Year, t in Technology}: OperatingCost[y,t]*DiscountFactorMoY[y,t] =
DiscOperatingCost[y,t];
#
#
################                Total Discounted Costs                  ############
#
#
s.t. EQ_TotalCost{y in Year, t in Technology}: DiscOperatingCost[y,t]+DiscCapitalInvestment[y,t] -
SalvageValue[y,t]+ AnnualTechEmPen[y,t] =TotalDiscCost[y,t];
/* Could do the technology summation here!!!*/
#
#
################                Investment Constraints                  #############
#
s.t. EQ_TotalAnnMaxCapacity{y in Year, t in Technology}: TotCapAnn[y,t] <=
TotalAnnMaxCapacity[y,t];
s.t. EQ_TotalAnnMinCapacity{y in Year, t in Technology}: TotCapAnn[y,t] >=
TotalAnnMinCapacity[y,t];
#
#
#
################                Reserve Margin                          #############
#
#
s.t. RM_TechIncluded{y in Year, l in LoadRegion}: sum {t in Technology}       TotCapAnn[y,t] *
RMTagTech[y,t] = TotCapinRM[y];
s.t. RM_FuelIncluded{y in Year, l in LoadRegion}: sum {f in Fuel}             Demand[y,l,f] *
RMTagFuel[y,f]  = DMDinRM[y,l];
s.t. RM_Constraint{y in Year, l in LoadRegion}:                               DMDinRM[y,l] * RM[y]
        <= TotCapinRM[y]*OneMatrixLR[l];
#
#
#################               Emissions                               #############
#
#
s.t. EmissionProduction1{y in Year, t in Technology, e in Emission}: sum{l in LoadRegion}
EmissionActvtyRatio[y,t,e]*Activity1[y,l,t]*YearSplit[y,l]=EmissionsProd1[y,t,e];
s.t. EmissionProduction2{y in Year, t in Technology, e in Emission}: sum{l in LoadRegion}
EmissionActvtyRatio[y,t,e]*Activity2[y,l,t]*YearSplit[y,l]=EmissionsProd2[y,t,e];
s.t. EmissionProduction3{y in Year, t in Technology, e in Emission}: EmissionsProd1[y,t,e] +
EmissionsProd2[y,t,e] = AnnualTechEmission[y,t,e];
s.t. EmissionsPenalty1{y in Year, t in Technology, e in
Emission}:AnnualTechEmission[y,t,e]*EmissionsPenalty[y,e]=AnnualTechEmPenByEm[y,t,e];
s.t. EmissionsPenalty2{y in Year, t in Technology}: sum{e in Emission} AnnualTechEmPenByEm[y,t,e] =
AnnualTechEmPen[y,t];
s.t. EmissionsPenalty3{y in Year, t in Technology}: AnnualTechEmPen[y,t]*DiscountFactorMoY[y,t] =
DiscAnnTechEmPen[y,t];#
```

Basic equations (1)

Objective function - minimize:

$$\sum_{y}\sum_{t} TotalDiscCost_{y,t}$$

$$TotalDiscCost_{y,t} = DiscOperatingCost_{y,t} + DiscCapitalInvestment_{y,t} - SalvageValue_{y,t} + DiscEmissionsPenalty_{y,t}$$

$$DiscOperatingCost_{y,t} = OperatingCost_{y,t} * DiscountFactor_{y,t}$$

$$OperatingCost_{y,t} = TotCapAnn_{y,t} * FixedCost_{y,t} + \sum_{l} (Activity_{y,l,t} * VariableCost_{y,t} * YearSplit_{y,l})$$

$$DiscCapitalInvestment_{y,t} = CapitalCost_{y,t} * NewCap_{y,t} * DiscountFactor_{y,t}$$

$$Production_{y,l,f} \geq Demand_{y,l,f} + Use_{y,l,f}$$

Where:

$$Production_{y,l,f} = \sum_{t} (Activity_{y,l,t} * OtptActvtyRatio_{y,t,f} * YearSplit_{y,l})$$

$$Use_{y,l,f} = \sum_{t} Activity_{y,l,t} * InptActvtyRatio_{y,t,f} * YearSplit_{y,l})$$

Basic equations (2)

$$\sum_{l} (Activity_{y,l,t} * YearSplit_{y,l} * EmissionActvtyRatio_{\text{¿}}) = EmissionsProd_{y,t,e}$$

$$EmissionsProd_{y,t,e} * EmissionsCost_{y,e} = EmissionsPenalty_{y,t,e}$$

$$\sum_{e} (EmissionsProd_{y,t,e} * EmissionsCost_{y,e}) = EmissionsPenalty_{y,t}$$

$$EmissionsPenalty_{t,y} * DiscountFactor_{t,y} = DiscEmissionsPenalty_{t,y}$$

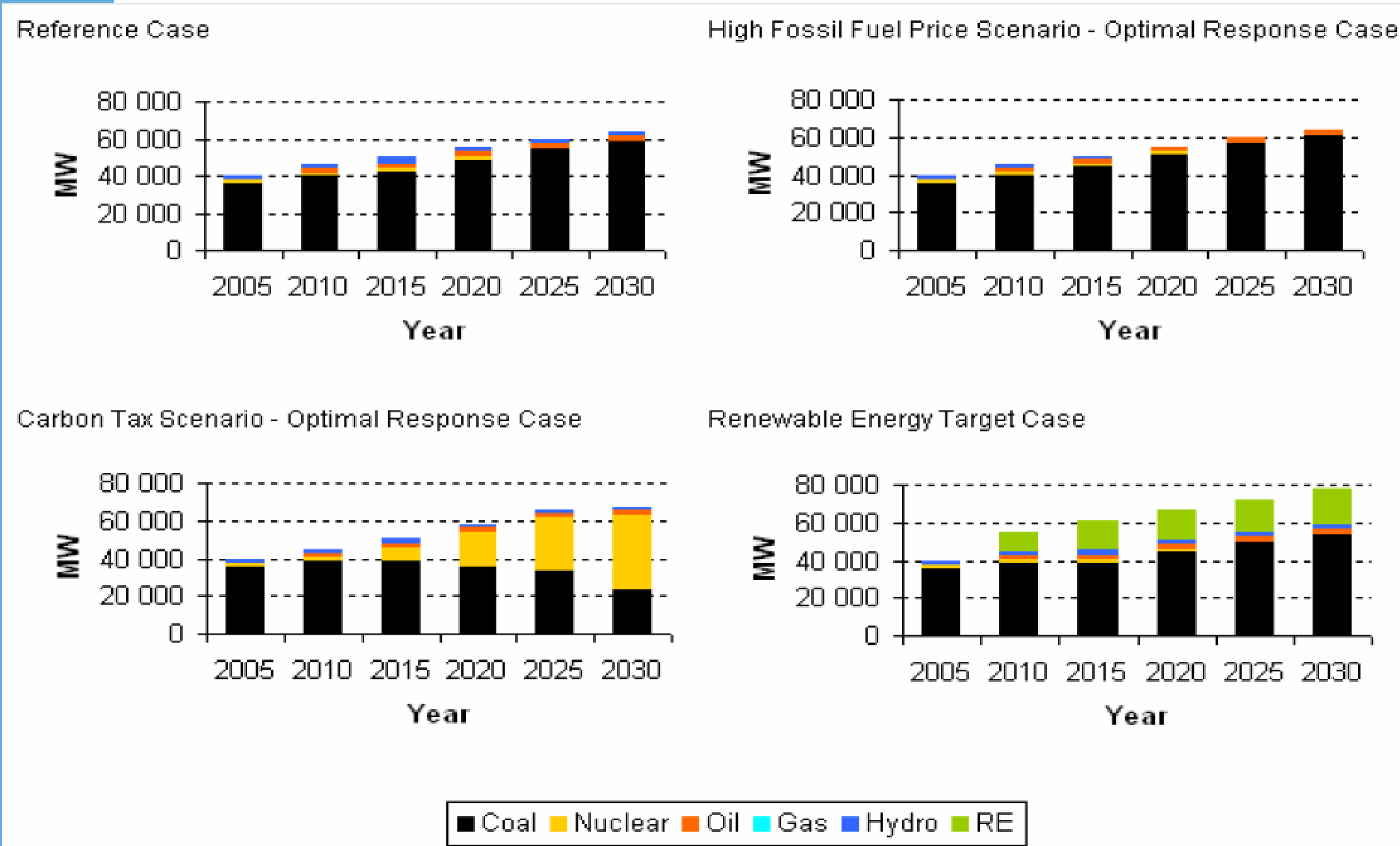
$$(DMDinRM_{y,l,f} + Use_{y,l,f}) * (1 + RM_{y,f}) \leq TotCapinRM_{y,f}$$

A Case study: High fossil fuel price imports, a carbon tax and a RE target

Repeated from Rogner et al WEC 2007

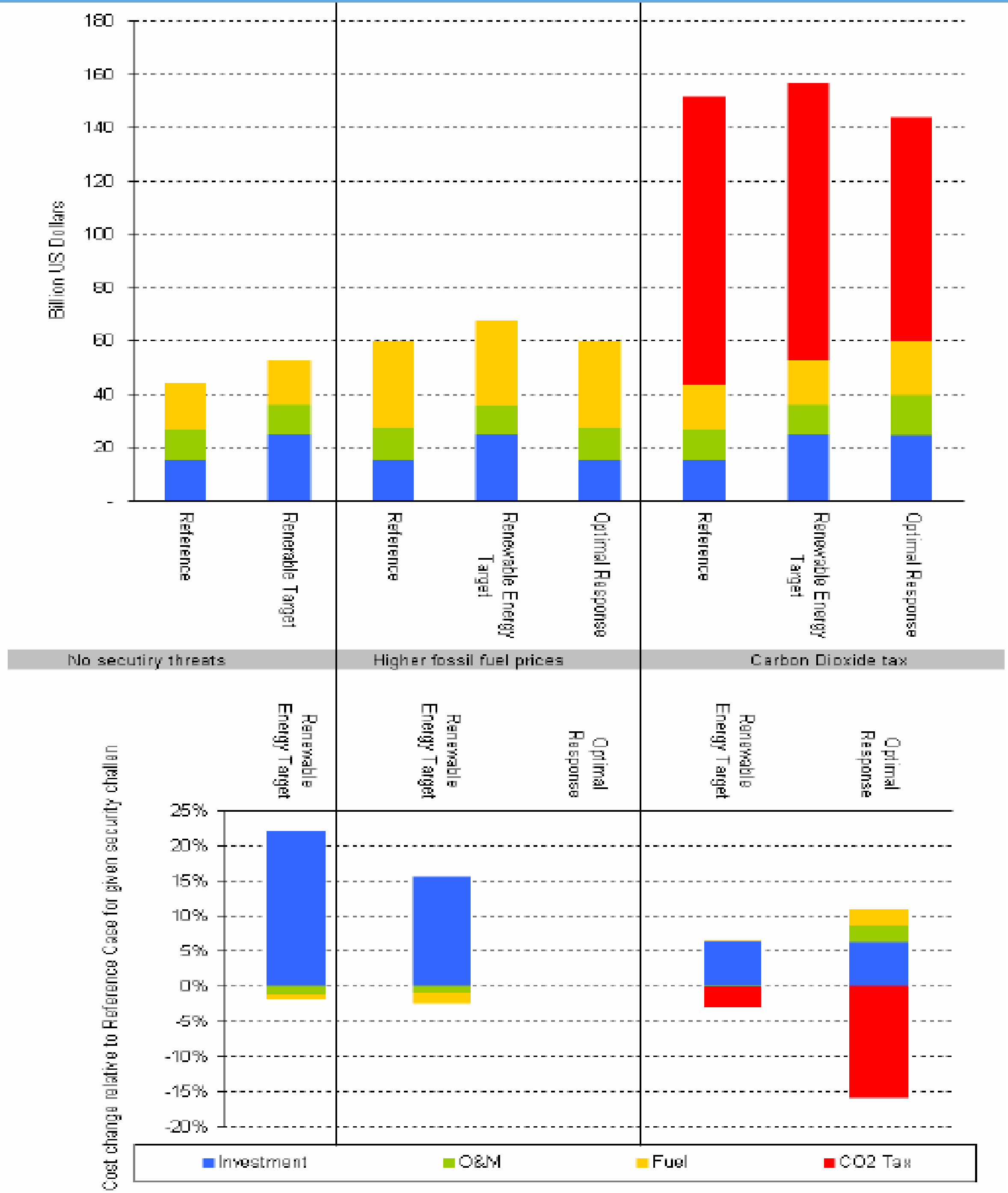
Security scenarios	**Cases**		
No security challenge	Reference case	Renewable energy target case (RET)	
High fossil fuel prices	Reference case under high fossil fuel prices	RET case under high fossil fuel prices	Optimal Response (OR) Case to high fossil fuel prices.
Carbon tax	Reference case under a carbon tax	RET case under a carbon tax	OR Scenario under a carbon tax.

Case Study: Total installed capacity



Case study:

- Total costs (top)
- Costs relative to Reference scenario
  - (under the same “security challenge”)



Conclusions

- Developed a first free OS energy planning model
- Hope to reduce the barriers to entry and increase modeling capacity at two levels:
  - Users
  - Developers
- Simplified
  - "Power modelers" will still need "power tools", solvers etc.
  - Not aimed at long term modelling (rather medium term annual granulation)
  - Applicable for a large set of "fist order" country applications
- Will continue development of components
  - Stand alone
  - Testing and deployment
  - Integration with other tools e.g. LEAP
  - This is an evolving tool (and a first iteration is presented here)
- Allows a basis for further development and experiments
  - Test new approaches on a simple small scale
  - Develop skills in research, university and other contexts
  - Come up with better representations than those given here