

# Extended MARKAL: A Brief User Manual for its Stochastic Programming and Multi-Region Features\*

**Amit Kanudia**

GERAD, Montréal, Canada

*E-mail: amit@crt.umontreal.ca*

**Richard Loulou**

McGill University and GERAD, Montréal, Canada

*E-mail: loulou@management.mcgill.ca*

*GERAD Working paper G-97-11*

March 1997

---

\* This work was supported by grants from Environment Canada, and Shastri Indo-Canadian Institute

The purpose of this document is to introduce the potential users to the new features of Extended MARKAL, stochastic programming and multiple region capability, developed at GERAD (Group for Research in Decision Analysis). It is also meant to help the new user in using the stochastic and/or multi-region features of the model and in producing tables and reports of his/her choice. Familiarity with the standard MARKAL and MUSS has been assumed. At present, the software operates through the FoxPro (2.0) command window. The graphs are plotted by a VB module in EXCEL (7.0). This document comprises two parts: Part A describes the operation of stochastic programming feature to include multiple scenarios, and the multi-region capability. The report writer is described in Part B.

Nous désirons présenter aux utilisateurs éventuels les récents développements effectués au GÉRAD (Groupe d'études et de recherche en analyse des décisions) en ce qui concerne les nouvelles caractéristiques du modèle Extended MARKAL: (i) la programmation stochastique et (ii) la possibilité d'étudier un système à régions multiples. Nous expliquons comment utiliser ces nouveaux outils de modélisation ainsi que la procédure à suivre pour produire les rapports désirés. Nous vous supposons familier avec la version standard de MARKAL ainsi qu'avec l'utilitaire MUSS. Le logiciel fonctionne présentement sous la fenêtre commande de FoxPro (2.0) Les graphes sont créés via un module VB d'EXCEL (7.0). Ce document est séparé en deux parties: la partie A décrit l'inclusion de scénarios multiples en programmation stochastique ainsi que la possibilité d'une étude multi-région. La partie B décrit le générateur de rapport.

## Part A

### Stochastic Programming in Extended MARKAL

#### Introduction

At the current stage of development, any combination of demand and carbon mitigation uncertainties (as a cumulative emission constraint or bifurcating tax trajectories) can be incorporated in an existing MARKAL model. The existing model has to be compatible with the Extended MARKAL of Canada (Berger and Loulou, 1993), which is the base model used for this development. Making an existing standard MARKAL model involves just a few database changes which require a few minutes of work.

The OMNI code has been modified to generate a stochastic model, but all the required data is captured through new tables that have been added within MUSS. The key change is the inclusion of a scenario index in the name of each variable and constraint. The typical modifications made in the OMNI code are shown in Box 1. Table 7 and Table 8 list the variable and constraint names in the OMNI code. Further, a utility called the *Scenario Generator* creates some of the new tables and helps in data input for scenario parameters. The report writer processes the text output from the optimizer and creates the standard analysis tables.

#### Using the Stochastic Feature

A new directory \SMARKAL should be created to unzip SMARKAL.ZIP<sup>†</sup>. Unzipping this file would install all the files needed to generate the stochastic program. The model and path specification in ..UM should be changed to SMARKAL. Depending on the optimizer, an appropriate command line must be added in ..UC to obtain a text file of the solution. For example TOASC in the case of Xpress. The use is described as a 5 step procedure:

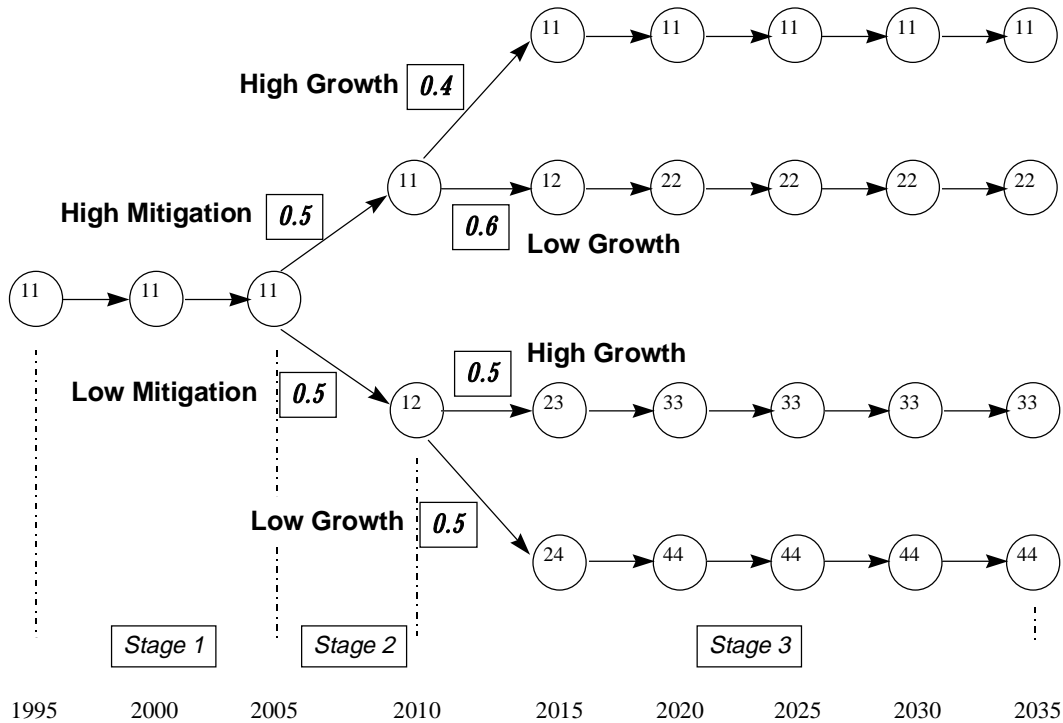
1. Construct an event tree (Figure 1) with the necessary scenario assumptions. Allot a two digit code to each node as per the following rule: for any node, the second digit is the serial order of that node among all nodes in a particular time period; and the first digit is the serial order of the parent node.
2. Create *Scenario Index Tables* in MUSS, shown in Table 1, Table 2, Table 3, and Table 4. One table, S(T), for each time period has the two digit codes of the nodes in the first column and the corresponding probabilities in the second column. The second column is titled PRBILITY and the values should add up to 1 for each table.
3. Now the Scenario Generator can be used to create tables CHILD(T,S(T)) and SARRAY(T). These have no additional information over the scenario index tables, but they are organized in a special format which is used by the OMNI code to generate the LP. The user is not required to know any details of these tables.
4. Create the tables SDEMD for capturing the demand scenario data. The scenario generator is particularly helpful in this because number of rows = (the number of demand categories  $\times$  number of scenarios). Table SDEMD has rows with names (DM)(C), where C is the set of scenarios (Table 5). So each scenario can have a different demand trajectory, specified as a multiplicative factor to be used with the base demand specified in DM.DEMAND. For example, if the demand for residential

---

<sup>†</sup> The authors can be contacted for this file.

appliances (RA) is 20% higher in period 9 under scenario 1, then the cell in column 9 and row RA1 will read 1.2. In addition to the table SDEMD, there has to be a STOCDEM=1 entry in the table CONSTANT.

5. Create the table SENV(ENV) to capture the mitigation scenario data. Table SENV(ENV) has rows CUM(C), which have the cumulative emission limits (Average annual limit  $\times$  number of periods) as TID entries (Table 6).



**Figure 1 Event Tree for Stochastic MARKAL**

### Performing a Standard MARKAL Run

1. Define the tree as a straight line, i.e., tables S1 to S9 have single rows with names '11' and probability = 1.
2. Put STOCDEM = 0 in the table CONSTANT. This disables the use of table SDEMD.
3. Relax the emission constraint in table SENV(ENV).

### Remarks

1. Each time the Scenario Generator is used to create tables, the *pack databases* utility in MUSS (..UXQQ) should be used once.
2. The model can be run from MUSS if the optimizer can handle the bigger problem. Otherwise the MPS code generated in file PROB.MAT can be submitted to some other optimizer (for example CPLEX). Once the solutions are ready in text files, the multi-scenario report writer (described in part B) can be invoked to compile the results.

**Table 1 Table S2: Stochastic Scenarios for Time Period 2**

	<b>PRBILITY</b>								
11	1.0								

**Table 2 Table S4: Stochastic Scenarios for Time Period 4**

	<b>PRBILITY</b>								
11	0.5								
12	0.5								

**Table 3 Table S5: Stochastic Scenarios for Time Period 5**

	<b>PRBILITY</b>								
11	0.2								
12	0.3								
23	0.25								
24	0.25								

**Table 4 Table S6: Stochastic Scenarios for Time Period 6**

	<b>PRBILITY</b>								
11	0.2								
22	0.3								
33	0.25								
44	0.25								

**Table 5 Table SDEMD: End-use Demand Scenarios**

	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>
RA1	1.00	1.00	1.00	1.00	1.15	1.17	1.18	1.19	1.20
RA2	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
RA3	1.00	1.00	1.00	1.00	1.15	1.17	1.18	1.19	1.20
RA4	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
RB1	1.00	1.00	1.00	1.00	1.15	1.17	1.18	1.19	1.20
---									

**Table 6 Table SENVCO2: Carbon Mitigation Scenarios**

	<b>TID</b>								
CUM1	320								
CUM2	320								
CUM3	555								
CUM4	555								

## Box 1 OMNI code modifications for Stochastic MARKAL

### Single Period Constraints And OBJ Values

```
FORM SECTION (TCH)(T)(SI)
FOR SI = TABLE S(T) (,1)
```

```
FORM VECTOR (T)(TCH)INV(SI;M2-2)
```

```
(T)ENV(ENV)(SI;M2-2) = TABLE ENV(ENV)..
```

```
PRICE = ..* TABLE S(T) (PRBILITY,(SI))
```

```
PRICES(C) = ...
FOR C = TABLE SARRAY (,1)
WHEN TABLE SARRAY(T)((SI;M2-2),(C))
```

### Multi-Period Constraints

```
(T/C)CPT(TCH)(CI;M2-2) = - 1..
FOR #C = MIN (($T + TABLE LIFE(TID,(TCH)) - 1), #T)
WHEN ($C .GE. $T)
WHEN ($C .GE. ($T + TABLE TCH(TCH) (TID,LAG)))
FOR CI = TABLE S(C) (,1)
WHEN TABLE CHILD(T)(SI)(C,(C)(CI))
```

## Box 1 OMNI code modifications for Stochastic MARKAL (Continued)

### Mitigation Scenarios

```
FORM VECTOR (T)EM(EEE)(SI;M2-2)
  FOR SI = TABLE S(T) (,1)
```

```
(T)ENV(EEE)(SI;M2-2) = -1
```

```
TOT(EEE)(C) = 1
  FOR C = TABLE SARRAY (,1)
  WHEN TABLE SARRAY(T)((SI;M2-2),(C))
```

```
FORM VECTOR REST
```

```
TOT(ENV)(C) = TABLE SENV(ENV)(TID,CUM(C))
  FOR C = TABLE SARRAY (,1)
  WHEN TABLE SENV(ENV) (,)
```

### Demand Scenarios

```
FORM VECTOR (T)(DMD)CAP(SI;M2-2)
(T)DEM(DM)(SI;M2-2) = TABLE CAPUNIT(TID,(DMD)) *...
```

```
FORM VECTOR REST
```

```
(T)DEM(DM)(SI;M2-2) = TABLE DM(DM)((T),DEMAND) *
  TABLE SDEMD((T),(DM)(SI;M2-2))
  FOR SI = TABLE S(T) (,1)
  EXCEPT DM = DM
```

**Table 7 Variable Names in Stochastic MARKAL**

<b>OLD NAME</b>	<b>NEW NAME</b>
<b>Annual activity of source (src) for energy carrier or material (enc)</b> (T)(SRC)ENC	(T)(S)(ENC)(I)‡
<b>stockpile activity in last period</b> Z(SRC)ENC	Z(S)(ENC)(I)
<b>investment of technology (tch) at period (t)</b> (T)(TCH)INV	(T)(TCH)INV(I)
<b>elastic markal</b> (T)00(DM)(K) (T)01(DM)(K)	(T)LE(DM)(K)(I) (T)GR(DM)(K)(I)
<b>installed capacity of demand-device (dmd) in period (t)</b> (T)(DMD)CAP	(T)(DMD)CAP(I)
<b>annual activity of process (prc) in period (t)</b> (T)(PRC)ACT	(T)(PRC)ACT(I)
<b>annual production of energy-carrier or material</b> (T)(PRC)(ENC)	(T)(PRC)(ENC)(I)
<b>maintenance of conversion technology (con) in season (z) at time period (T)</b> (T)(ELA)(Z)M	(T)(ELA)(Z)M(I)
<b>installed capacity of conversion technology (ela) in time period (T)</b> (T)(XXX)CAP	(T)(XXX)CAP(I)
<b>annual activity of conversion technology (ela) in season (z) and daytime (y) in time period (t)</b> (T)(XXX)(Z)D	(T)(XXX)(Z)D(I)
(T)(XXX)(Z)N	(T)(XXX)(Z)N(I)
<b>annual heat production of pass-out turbine (cpd) in season (z) and daytime (y) in time period (t)</b> (T)(XXX)(Z)DH	(T)(XXX)(Z)DH(I)
<b>installed capacity of heating plant (hpl) in time period (t)</b> (T)(XXX)CAP	(T)(XXX)CAP(I)

‡ (I) refers to the Scenario Order, i.e., the second digit in the Scenario Index; (C) refers to the scenario orders of the terminal period.

**annual production of low-temperature heat from heating plant (hpl) in season (z) in time period (t)**  
 (T)(XXX)(Z) (T)(XXX)(Z)(I)

**activity of a component (opr) producing a (enc) at time (T)**  
 (T)B(BLE)(OPR) (T)(BLE)(OPR)(I)

**new Vector (T)EM(ENV) for emissions account/constrain**  
 (T)EM(EEE)(I)

**discounted cost of stochastic scenarios**  
 PRICEST(C)

**Table 8 Constraint Names in Stochastic MARKAL**

<b>OLD NAMES</b>	<b>NEW NAMES</b>
<b>objective function for price - scenario</b> PRICE = OBJ	PRICE
<b>discounted cost of stochastic scenarios</b> PRICES(C)	PRICES(C)
<b>objective function for price to reduce demand in markal-ed scenario</b> REDUCE= OBJ	REDUCE(C)
<b>objective function for price to grow demand in markal-ed scenario</b> GROW = OBJ	GROW(C)
<b>objective function for the first step of optimization for renewable scenario</b> PRICER = OBJ	PRICER
<b>annual system cost accounting</b> (T)ANNCOST = N	(T)ANNCOST(I)
<b>annual capital investment cost accounting</b> (T)INVEST = N	(T)INVEST(I)
<b>annual capital investment cost accounting</b> (T)ANNINV=N	(T)ANNINV(I)
<b>correction for unused life of investments</b> (T)SALVAGE = N	(T)SALVGE(I)
<b>objective function for the first step of optimization for security scenario</b> SECURITY = OBJ	SECURITY

(T)SECURTY = N	(T)SECURITY(I)
<b>energy-carrier balance</b> (T)BAL(ENC) = N	(T)BAL(ENC)(I)
<b>energy-carrier or material cumulative constraint</b> CUM(SRCENCP;M1-1)(SRCENCP;M4-7) = L	CU(SRCENCP;M1-1)(SRCENCP;M4-7)(I)
<b>energy-carrier or material growth constraint</b> (T/U)GR(SRCENCP;M1-1)(SRCENCP;M4-7) = L	(T/U)G(SRCENCP;M1-1)(SRCENCP;M4-7)(I)
<b>technology capacity inter-period transfer</b> (T)CPT(PON) = E	(T)CPT(PON)(I)
<b>technology growth constraint</b> (T/U)GRC(TCH) = L	(T/U)GR(TCH)(I)
<b>constraint on growth of total investment costs</b> (T)SMOOTH = L	(T)SMTH(I)
<b>demand equation</b> (T)DEM(DM) = G	(T)DEM(DM)(I)
<b>process balance, variable-output process</b> (T)PBL(PRC) = E	(T)PBL(PRC)(I)
<b>limited energy-carrier or material output variable-output process</b> (T)(PRC)(ENC)L = L	(T)(PRC)(ENC)(I)
<b>process-utilization capacity constraint</b> (T)UTL(PRC) = L	(T)UTL(PRC)(I)
<b>scheduled maintenance for conversion technologies</b> (T)UTL(CON) = G	(T)UTL(CON)(I)
<b>conversion technology, bound constraint</b> (T)BND(CON)(BD;M1-1) = (RTY/BD)	(T)BN(CON)(BD;M1-1)(I)
<b>balance of electricity, seasonally and diurnally</b> (T)BALE(E)(Z)(Y) = G	(T)BLE(E)(Z)(Y)(I)
<b>electricity-peaking constraint, day only</b> (T)EPK(E)(Z)D = G	(T)EPK(E)(Z)D(I)
<b>electricity-production capacity constraint</b> (T)E(Z)D(ELA) = L	(T)E(Z)D(ELA)(I)

**electricity production, base-load constraint**

$$(T)BAS(E)(Z) = L$$

$$(T)BAS(E)(Z)(I)$$

**balance of district heating by season**

$$(T)BALDH(H)(Z) = G$$

$$(T)BADH(H)(Z)(I)$$

**district-heating peaking constraint, only winter**

$$(T)HPKW(H) = G$$

$$(T)HPKW(H)(I)$$

**district-heating production capacity constraint**

1 no peak device

2 peak device ('tid'-entry in table peak)

$$1.(T)HP(Z)(HPL) = L$$

$$(T)HP(Z)(HPL)(I)$$

$$2.(T)HPW(HPL) = L$$

$$(T)HPW(HPL)(I)$$

**user defined constraints**

$$(T)(ADRATIO) = (RTY)$$

$$(T)(ADRATI)(I)$$

**annual reservoir management capacity constraint**

$$(T)ARM(HDE) = L$$

$$(T)ARM(HDE)(I)$$

**seasonal reservoir management capacity constraint**

$$(T)SRM(Z)(HDE) = L$$

$$(T)SR(Z)(HDE)(I)$$

**blending constraints of (enc)**

$$(T)(BLE)(SPE)L = L$$

$$(T)(BLE)(SP)L(I)$$

$$(T)(BLE)(SPE)G = G$$

$$(T)(BLE)(SP)G(I)$$

$$(T)(BLE)(SPE)E = E$$

$$(T)(BLE)(SP)E(I)$$

$$(T)(BLE)(SPE)N = N$$

$$(T)(BLE)(SP)N(I)$$

## Multiple Regions in Extended MARKAL

### Introduction

The Extended MARKAL can readily combine multiple MARKAL models as a multi-region MARKAL. The user may allow inter-regional exchanges of emission permits (through a joint emission target) and/or of various energy forms.

This feature works as follows: The MPS files<sup>§</sup> from various regions are used as input by a FoxPro program to generate the multi-region LP. First, a region index is added to the variable and constraint names in each of the MPS files. Then each MPS file is distributed into four files: Rows, Columns, RHS, and Bounds. Individual objective functions are transferred to variables and their sum is used as the new objective function. Appropriate constraints are added to link the EXPENCP(RI) and IMPENCP(RI) variables. RI is the regional index. Variables and constraints are added to replace individual emission constraints with joint ones. All this is controlled by simple parameter declarations.

### Using the Multi-region Feature

Generating the multi-region LP is a single step process, controlled by the parameters declared in two .DBF files. The first file is REGIONS.DBF (Table 9). Here, the user declares the various regions that are to be combined. For example, to combine four regions, this file will have four records, each of which will have a common name in the field OPFILE.

**Table 9 Structure And Functioning Of File REGIONS.DBF**

FIELD NAME	DESCRIPTION	FUNCTION
MARK	Mark	If there is a "*" in this field, the record is ignored
REGION	MUSS database Name	
INDX	Region Index	This character is added to all variable and constraint names in the MPS file
OPFILE	Output File	The joint MPS file is written under this name. Extension .MPS is always added

*Energy Trading Declarations:* Energy trading is controlled by very simple declarations in the file TRADES.DBF. It has a single field, NAME, which takes entries of the following form: (ENC)(P)(RI1)(RI2). For example, as a result of the declaration "ELC2OQ", Electricity can be exported from Ontario (regional index O) to Québec (regional index Q). The 'P' index is used to have the EXP/IMP tables in MUSS. Bounds or costs can be declared in the table EXPELC2 of Ontario, or IMPELC2 of Québec.

The MPS file thus generated can be submitted to a solver. The solution, as a text file, can be analyzed with the multi-region report writer described in part B.

---

<sup>§</sup> There is no special requirement for the MPS files except that they should have the IMPENCP and EXPENCP tables corresponding to the inter-regional energy exchanges. One 'P' index is to be fixed for each exchange. For example, for a two-way exchange of electricity, both Québec and Ontario have tables IMPELC2 and EXPELC2.

## Part B

### The Multi-Scenario Multi-Region Report Writer for Extended MARKAL

#### Introduction

The report writer described here fully supports the multi-scenario and multi-region features of Extended MARKAL. It takes the text output from the solver as the input and processes it in two stages, as shown in Figure 2. The double lined boxes indicate text files, simple boxes are database files and the shaded boxes are user defined parameters that control the report generation. The first stage compiles a set of pre-defined standard tables for each of the scenario/region. This stage compiles very detailed results including fuel consumption and emissions for each technology. The second stage produces the final tables/graphs, based on the user specified tables. The user has considerable control over format and presentation of the results. For example, the order in which the tables are printed, type and size of graphs etc. The tabular output uses FoxPro and the graphs are produced in EXCEL. An important advantage of this system is that it produces the results in WINDOWS compatible format. This system is operated from the FoxPro command window, and the various parameters are specified in .DBF files. It is possible to build a visual basic interface for this system in 6 to 8 weeks time.

Detailed description of the two stages, along with illustrative examples, follows.

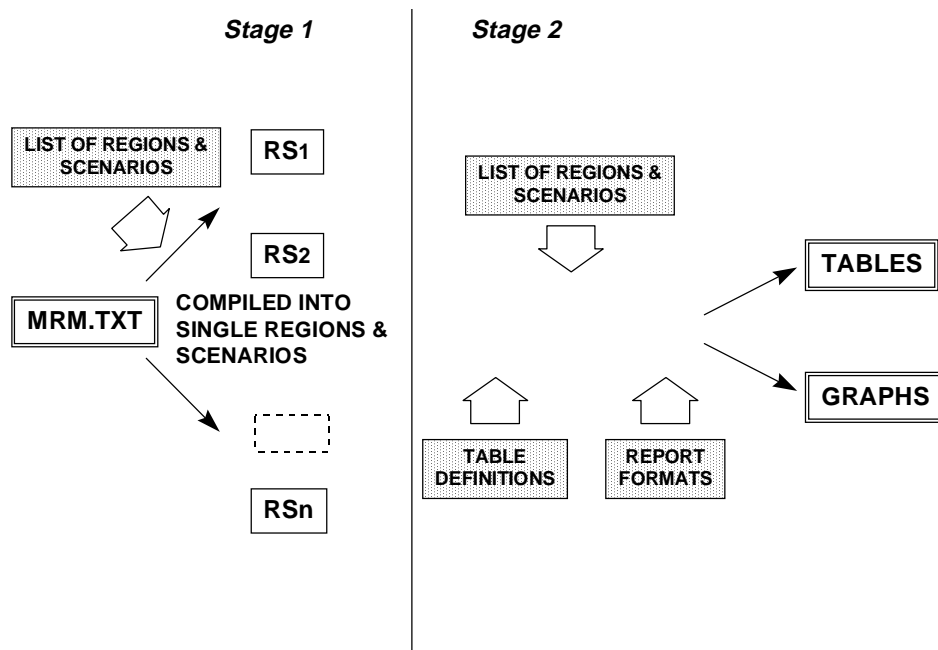


Figure 2 Data Processing In The Multi-Scenario Multi-Region Report Writer

### First Stage: Standard Tables Compiler

The first stage of the report writer extracts the different scenarios and/or regions embedded in the solution from a single run. The different scenarios are identified based on the event tree. The region index added to the variable names is used to separate the results region wise. This is controlled by the parameter file **SCENDECS.DBF**. A description of the fields and their function is given in Table 10. More than one solution file can be compiled by listing them in this file. Note that different scenarios are represented by a single record and are generated with names as SCENCASE(n). Whereas, each region is represented by a different row (record) in this file.

*Remark 1: Some of the information here is common with that specified for combining different models to form a multi-region model. This is because the report writer may be used for a single region as well. The repetition will be taken care of when an interface is made.*

**Table 10 Structure And Functioning Of File SCENDECS.DBF**

FIELD NAME	DESCRIPTION	FUNCTION
MARK	Mark	If there is a '*' in this field, the tables are not generated for the corresponding scenario
SCENCASE	Scenario Case	Name with which the results are to be stored; usually 'solfile + rindx' is convenient
SOLFILE	Solution File	Name of the file which has the text output from the solver
INDX	Scenario Index	This is used to specify the scenario number when compiling the results for a perfect foresight strategy in a multiple scenario run.
RDBS	Region Database	Name of the MUSS database as in \MUSS\DBF\?
STREE	Scenario Tree	Name of the file which has the MUSS tables S(1) to S(T) corresponding to the run

*Remark 2: The information on the event tree could be gathered from the MUSS database directly. It was preferred to store it in a separate table so that it is not compulsory to generate the reports for a particular scenario before the tree is modified in any way.*

The various standard tables produced in this stage are described in **Table 11** and **Table 12**.

**Table 11 The Standard Tables In Report Writer**

<b>TABLE NAME</b>	<b>DESCRIPTION</b>	<b>ROW COMPONENTS</b>
<b>CAP</b>	Technology-wise Capacity	.(TCH)
<b>INV</b>	Technology-wise Investment	.(TCH)
<b>ACT</b>	Technology-wise Activity	.(TCH)
<b>SUPPLY</b>	Fuel-wise Supply (Only the first character of SRC)	.(S)(ENC)(P)
<b>ELCGEN</b>	Technology-wise Electricity Production	.(ELE)(Y)(Z)
<b>T08</b>	Technology-wise Fuel Consumption	(ENC).(TCH)
<b>T09</b>	Dual Values.	.(DM) .(ENC)
	Electricity: "Average" over all Y and Z - Peaking Constr. + Balance Constr	.AVG
	Electricity: Balance constr. (Y,Z) Dual Value * 'fraction of year'	.(Y)(Z)B
	Electricity: Peaking constr. (Y,Z) Dual Value * 'fraction of year'	.(Y)(Z)P
<b>T01</b>	Expected Value of Objective function	PRICE
	Scenario Objective function Value	PRICST
	Annual System Cost Period wise Investment Cost Period wise salvage value Annualized Investments Annual Welfare loss	ANNCOST INVEST SALVGE ANNINV ANNWELF
<b>EMISSION</b>	Aggregate Annual Emission	.(ENV)
<b>(ENV)</b>	Technology-wise Emission Tables	.(TCH)(ENV)
<b>ELAST</b>	Demand Loss (DM Units) in MARKAL-ED	.00(DM)(K)
	Demand Increase (DM Units) in MARKAL-ED	.01(DM)(K)
<b>ELASP</b>	Demand Loss (%) in MARKAL-ED	.00(DM)(K)
	Demand Increase (%) in MARKAL-ED	.01(DM)(K)

All tables have time periods as columns except for two rows of T01 shown below:

**Table 12 Columns for the PRICE and PRICEST rows in T01**

	<b>Tech+Fuel</b>	<b>Welfare Loss</b>	<b>System Cost</b>	<b>Carbon Tax</b>	<b>Obj. Value</b>
<b>PRICE</b>					
<b>PRICST</b>					

## Second Stage: Custom Tables

The second stage compiles information from the standard tables based on the custom tables defined by the user. Two classes of custom tables can be defined: i) Detailed scenario results, where the rows represent solution values for the same scenario, and ii) Comparative scenario results, where each row represents a different scenario. **Each region has its own set of custom table files.**

The custom table definition is very flexible because of a very special scheme of *identifying* and *transferring* data from the standard tables. The declarations for the two classes of tables are slightly different. Declaration for tables of Class 1 has an additional step due to declarations relating to aggregation and naming of rows. Class 1 declarations are described first.

### *Detailed Scenario Results custom tables*

For each table, list of templates of up to 3 characters can be specified to identify rows from a particular standard table. The position in the variable name, where the first character of the template has to match, can be specified. Technologies matching with the template can be (optionally) tested for being members of class DMD or PRC. While transferring, the row can be given a name that is a different part of the row name in the standard table. For example, consider compiling fuel consumption in the transport sector. Rows can be identified in standard table T08, with template “T\$\$”, beginning at the 5<sup>th</sup> character and tested for being member of class DMD. Using the OFSET and LNTH fields, the rows can be transferred with the first three characters as the name. See example in Table 14.

Regarding naming the rows, there are two options: 1. If the existing name appears as a member name in some MUSS class, that name can be used by default (leave field GRP blank, Table 14), and 2. If a special name is required or there is some aggregation to be done, for example WOO+WOR+WOP has to be called as Biomass, Then the file <RDBS>\_GRP.dbf can be used as shown in the example (Table 15 and Table 16). **All rows having the same name are always added.**

Each table is identified by a unique 5 character code. The selected table are sorted on this code in the descending order. The detailed scenario tables are defined in files <RDBS>\_RPT.dbf. The structure and functioning of this file is described in Table 13.

### *Comparative Scenario Results custom tables*

These tables are declared in files <RDBS>\_CMP.dbf (structure and function described in Table 17). The specifications are a subset of those for the detailed scenario tables. Examples are given in Table 18 and Table 19.

**Table 13 Structure And Functioning Of File <RDBS>\_RPT.DBF**

<b>FIELD NAME</b>	<b>DESCRIPTION</b>	<b>FUNCTION</b>
MARK	Mark	If there is an '*' in this field, the table is not included in the report
DM	Table Code	A unique 5 character code for each table
TABL	Standard Table	Name of the standard table from which data is to be extracted
GRP	Grouping	A "G" here implies that the file <RDBS>_GRP.DBF has information about naming the rows
CRP	Composite Reports	An "*" in this field eliminates this table from composite reports
G	Graph Type	'L'=> Line; 'A'=> Area; 'B'=> Bar
DML	DMD List	List of templates which belong to class DMD
PRL	PRC List	List of templates which belong to class PRC
GENL	General List	List of templates not to be checked for any class membership
MINL	Minus List	List of templates to be excluded from the identified rows.
STRT	Start	Position in the row name of standard table where first character of the template should match
OFFSET	Off-set	Off-set of the beginning of the character string in standard table row name, to name the transferred row (STRT as 0)
LNTH	Length	Length of the character string in standard table row name, to name the transferred row
SOC	Secondary Condition	Another template that the name (transferred) has to match
MULTFACT	Multiplying Factor	Factor with which the entire row is to be multiplied
PLU	Plus	More than one rows can refer to the same table if there is a "+" in this field for all subsequent rows.
TITLE	Title	Table title, to appear on tables and graphs
UNITS	Units	The units to appear below the title in tables and on the Y Axis in Graphs

<b>EXAMPLES</b>
-----------------

**Table 14 Table Declarations for Fuel Composition in Transport Sector: File qc\_rpt.dbf**

DM	TABL	G	GRP	STRT	OFSET	LNTH	PLU	MULT-F	DML	MINL
JJ01	T08	A		5	4	3			T\$\$	

**Table 15 Table Declarations for Aggregate Primary Energy: File qc\_rpt.dbf**

DM	TABL	G	GRP	STRT	OFSET	LNTH	PLU	MULT-F	GENL	MINL
ZZ01	SUPPLY	A	G	2	0	3		2.59	IEL	
ZZ01	ELCGEN			2	0	3	+	2.59	E3	E31 E33 E35
ZZ01	SUPPLY			2	-1	3	+		ING	
ZZ01	SUPPLY			2	-1	3	+		IPL ICR IDF	
ZZ01	SUPPLY			2	1	2	+	0.5916	ITH IUR	
ZZ01	SUPPLY			2	-1	3	+		RW	
ZZ01	SUPPLY			3	0	3	+		COA PCO	

**Table 16 Table Declarations for Aggregate Primary Energy: File qc\_grp.dbf**

DM	ROWLABEL	MEMBERS
ZZ01	Hydro Electricity	E3
ZZ01	Electricity Import	IEL
ZZ01	Biomass	WOD WOR WOO WOA WOP WPE
ZZ01	Liquids	N6\$ CR\$ DFO LPO DSL JTF
ZZ01	Nuclear (eq.)	.I
ZZ01	Natural Gas	NGA NGH
ZZ01	Coal	COA PCO

**Table 17 Structure And Functioning Of File <RDBS>\_CMP.DBF**

FIELD NAME	DESCRIPTION	FUNCTION
MARK	Mark	If there is an "*" in this field, the table is not included in the report
DM	Table Code	A unique 5 character code for each table
TABL	Standard Table	Name of the standard table from which data is to be extracted
CRP	Composite Reports	An "*" in this field eliminated this table from composite reports
G	Graph Type	'L'=> Line; 'A'=> Area; 'B'=> Bar
DML	DMD List	List of templates which belong to class DMD
PRL	PRC List	List of templates which belong to class PRC
GENL	General List	List of templates not to be checked for any class membership
MINL	Minus List	List of templates to be excluded from the identified rows.
STRT	Start	Position in the row name of standard table where first character of the template should match
SOC	Secondary Condition	Another template that the name (transferred) has to match
MULTFACT	Multiplying Factor	Factor with which the entire row is to be multiplied
PLU	Plus	More than one rows can refer to the same table if there is a "+" in this field for all subsequent rows.
TITLE	Title	Table title, to appear on tables and graphs
UNITS	Units	The units to appear below the title in tables and on the Y Axis in Graphs

**EXAMPLES**

**Table 18 Table Declarations for Electricity Consumption In Demand Category RA: File qc\_cmp.dbf**

DM	TABL	G	STRT	SOC	DML
CT1	T08	L	5	ELC	RA\$

**Table 19 Table Declarations for Aggregate Primary Energy: File qc\_cmp.dbf**

DM	TABL	G	STRT	PLU	MULT-F	GENL	MINL	REMARKS
CT2	SUPPLY	L	2		2.59	IEL		<i>Electricity Import</i>
CT2	ELCGEN		2	+	2.59	E3	E31 E33 E35	<i>Hydro Electricity</i>
CT2	SUPPLY		2	+		ING		<i>Gas Import</i>
CT2	SUPPLY		2	+		IPL ICR IDF		<i>Oil Import</i>
CT2	SUPPLY		2	+	0.5916	ITH IUR		<i>Nuclear Energy</i>
CT2	SUPPLY		2	+		RW		<i>Renewable Energy</i>
CT2	SUPPLY		3	+		COA PCO		<i>Coal Import</i>

## References

Berger, C. and Loulou, R. (1993), *Extended MARKAL: A Brief User Manual*, GERAD Technical Report, Montréal, Canada.