RadConEd: A Graphical Data Editor

for the

Radiological Consequences Model, RadCon

Version 1.0

J. Crawford
R.U. Domel

ansto

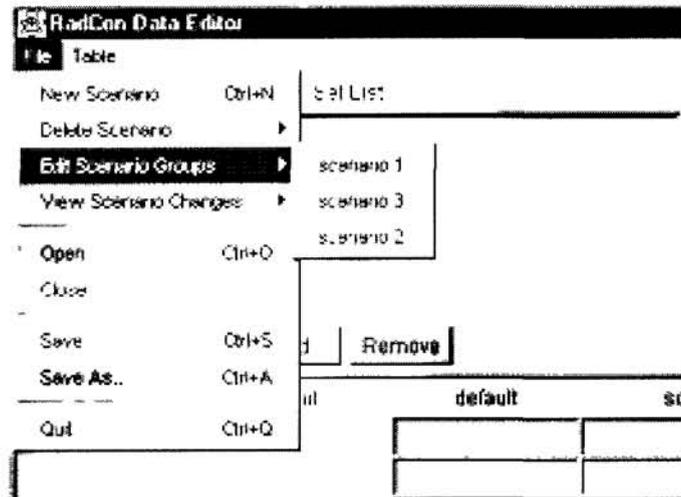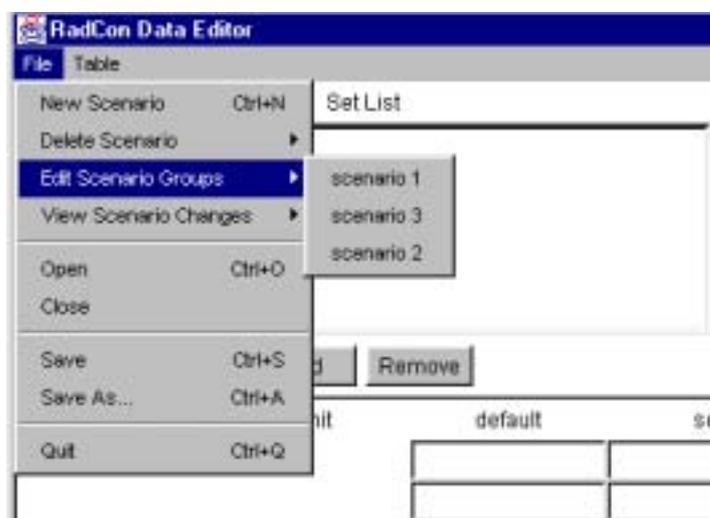Australian Nuclear Science and Technology Organisation

RadConEd: A Graphical Data Editor

for the

Radiological Consequences Model, RadCon

Version 1.0

J. Crawford
R.U. Domel

Prepared within the Radiological Consequences Project
May 2000

Australian Nuclear Science and Technology Organisation

Table of Contents

List of Figures

1 Introduction

The Radiological Consequences model, RadCon [1, 2], developed at the Australian Nuclear Science and Technology Organisation (ANSTO), requires values for a large number of parameters. A description of these parameters and their interpretation is given in [1]. Suitable values of the parameters are selected for each case study and stored in data files, which are accessed by the RadCon program during execution. These parameters have been grouped into separate files, based on their physical and biological functions. The data/parameters used for an Australian tropical/subtropical scenario are described in a separate report [3]. A further report [4] describes data/parameters for a particular temperate scenario.

This document describes the application, RadConEd, which has been designed and implemented to enable users of the RadCon system to update these parameter files. The RadCon system is written in the Java programming language, and as such provides portability across computer platforms. The software described in this report was developed in line with the portability requirements of RadCon, thus providing a uniform user interface across computer platforms and bypassing the need of using system editors. In addition a number of data integrity measures were implemented.

2 Installing & Running the Application

2.1 Installing the application

To install the RadCon data file editor you need to `unzip' the RadConEd.zip file to a directory. You can unzip the file with any 'zip' type compatible program (ie WinZip, pkunzip, unzip etc…). After you have unzipped the RadConEd.zip file, there should be several '.*class*' and '.*java*' files. The '.*java*' files have been included should any local modifications be required (see Appendix A if you need to re-compile the source code). In addition a file '*basedatafiles.zip*' should be available (see Appendix B). This file contains sample data files and is used as the basis for creating new scenarios.
Note: In cooperation with your computer support personnel, this step will have been implemented on your PC computer with the installation of RadCon and a shortcut displayed on the computer screen (possibly titled RadConEd).

2.1.1 File locations

The application can be run from any directory which contains all of the '.*class*' files as supplied in the RadConEd.zip file, or by using the Java CLASSPATH environment variable to point to the location of the class files. The application would normally be extracted to a directory at the same level as the RadCon application.
Note: If you intend to create new scenarios you need to ensure that the file '*basedatafiles.zip*' is available in your working directory.

2.2 Running the application

To run the application the JAVA™ Virtual Machine (JVM) is required. This application has been designed to run with a JAVA 1.1 compliant JVM. The standard AWT functions only are used, thus the system should work with any release of JAVA from 1.1 to 1.1.8.

To run the application under Linux or Unix systems, you can enter the following at the command prompt:

```
java –jit TableEdiotr [Return]
```

or set up a shell script, RadConEd, with the following content or equivalent:

```
#! /bin/sh
CLASSPATH=<location of class files>
export CLASSPATH
java -jit TableEdiotr
```

Note: If the shortcut has been installed for the Microsoft Windows ® platform, then double click on the shortcut to bring up the main screen of RadConEd. Figure 1 shows the main screen of RadConEd.

For simple data editing read section 3.1 then go to section 3.5

Figure 1: The RadConEd main screen.

3   The Data files & Meta-Data files
3.1   Underlying data files

The parameter data required by the RadCon program is stored in separate files, containing groups of related parameters. Each parameter has five values associated with it. These values are:

- the default value,
- the minimum of the parameter (*min*),
- the maximum value (*max*),
- the distribution type (*dist*), and
- the standard deviation (*sd*).

The default values are stored in the main file for that group of parameters with a preset *basefilename* eg AnimalInfo, CowDiet, Deposition_Info, etc. The file naming convention and the contents of the files are given in Appendix C with a sample data file given in Appdendix D. The remaining values (i.e. minimum, maximum, distribution type and standard deviation)

are stored in files whose names are created by prefixing the *basefilename* with a string, *min* for the minimum values, *max* for maximum values, *dist* for the distribution type and *sd* for the standard deviation (do not select these when editing data/parameter values, setect only the base file name). The format of these files has to mirror the file with the default values.

When using the RadConEd program you only specify the *basefilenam*e. This signals to the editor to read all the associated files, which must all have the information in the same order and format.

The TableEditor application uses a 'zip' archive as the method for keeping a set of 'template' data files. This file is to be called 'basedatafiles.zip' and is to be located in the current working directory. This can be used as the starting point for a new scenario.

3.2   Data files grouping

All the data files for one scenario are stored in the same directory. The data files for separate scenarios can be grouped into separate directories. The scenario name is used to name the directory for that scenario. In each of these directories there is a GROUPS.DAT file. This file defines what is contained in the actual data files. One cannot directly add new fields in the data files. New fields can only be added if they exist in the GROUPS.DAT file. The GROUPS.DAT file is broken into sections, which define '*Groups*' or types of data attributes, e.g. the SOIL group contains the types *sand*, *humus*, and *clay*. The file also defines which data files will contain the group and whether all of the attributes must appear in the file or just some of them. See Figure 3 for an example of the information stored in this file and Appendix E for a sample file format.

3.3   Editing the Groups for a scenario.

Each of the tables in the system is defined by data '*Groups*'. For each group there are a number of variables, and corresponding data files which contain instances of the group. The Group File Editor is where you can alter the GROUPS.DAT file that defines these groups. To open the Group File Editor, select the File menu followed by the '*Edit scenario groups*', see Figure 2. When you place the mouse over the '*Edit scenario groups*' option or click on it, all of the currently available scenarios will appear as a sub-menu. Click on one of the scenario names to open the Groups editor.

Figure 2: Editing the GROUPS.DAT file.

When the Group File Editor has loaded you will see the screen shown in Figure 3. From here you can create, edit or remove groups. To create a new group simply click the '*Add*' button which is located below the Groups List, a dialog will pop-up asking for a new group name, enter a new group name (no spaces). To create the group, simply click the '*ok*' button. A confirmation will pop-up on successful creation. The new group name will now appear in the groups list on the left-hand side of the screen. To change a group name, select it from the list by clicking on it, and then click the '*Edit*' button, a dialog will pop-up with the existing group name in the text field section, simply edit the name and then click the '*ok*' button or '*cancel*'. To remove a group select it from the list and then click on the '*Remove*' button below the Groups list. To edit a group you need to make a selection from the Groups list, then to add, edit or remove variables or files, use the buttons below the corresponding lists in the same manner as for the groups list buttons. The only difference is that for the files section extra dialog will pop-up asking for the type of file relation when adding/editing a file. The relation can be either ALL or ANY. Use ALL for when the file specified should contain all of the variables of the group, or ANY when it may only contain some of the variables. To save the changes made, click the '*Save*' button. To exit the Group File Editor, click the '*Quit*' button. This option will also prompt you to save changes made.
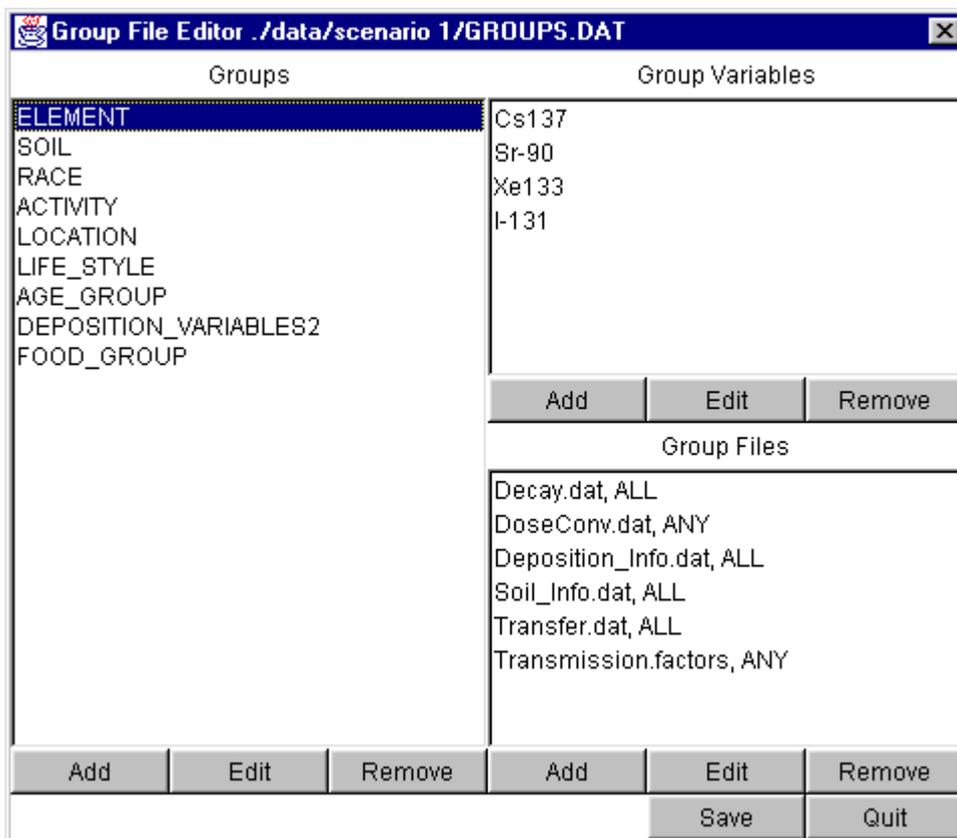
Figure 3: The GROUP.DAT file Editor.

3.4   Creating a new scenario

To create a new scenario, select the *File* menu option by clicking on it.   A drop down window will appear.   On selecting the *New Scenario* option a dialog window (see Figure Figure 4) will be displayed.   Enter the name for the new scenario followed by selecting the *Done* button.   The application will attempt to create a new directory for the scenario and then copy all of the template data files to the new scenario. The template date files are those

contained in the 'basedatafiles.zip' file. You will get a confirmation message of the success or failure of the create process.
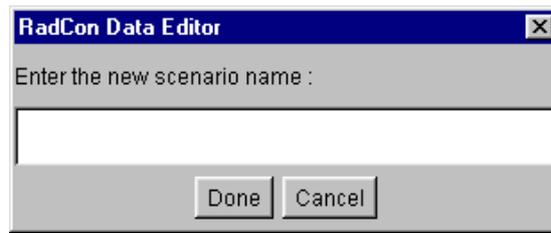
Figure 4: Creating a new scenario.

Note: Sample parameter values will be supplied with your RadCon Data Editor. You can create a new scenario using these values and editing them appropriately. If you wish to use template files other than those supplied in the distribution system, you will need to create a '*basedatafiles.zip*' file using the required data files and copy it in your work directory.

3.5   Editing a scenario data file.
To edit a data file (see Appendix C for a description of the data files) and the tables contained in the file you need to:
1. Open the data file using the *File* menu option. For example if using the default directory */data/tables* follow the following steps:
   - From the top menu bar select *File* and click on *Open*
    - Double click on the folder called *data*. A number of folders are contained here including the default data called tables and any generic or saved scenarios
    - Double click on the scenario you require – most likely it will be the generic tropical/subtropical scenario or one that has already been created and saved
2. Select the table within the data file, using the *Table* menu option,
3. Select the group to edit by selecting a row in the left rectangle of the main screen, see Figure 1. In this figure it is labelled *Group List*. This label will change, depending on the contents of the table being edited.
4. Select the variable to edit by selecting a row in the right rectangle of the main screen, see Figure 1. In this figure it is labelled *Variable List*. This label will change, depending on the contents of the table being edited. By the selection of this variable, the associated data (including sd, min, max, dist) will be listed in the table below the left and right rectangles.
5. Make the required changes.
6. Save the changes.
These functions are detailed in subsequent sections.

3.5.1   Opening and Saving a data file

To open a data file select the File menu followed by '*Open (File)*'. If a file is currently open and changes have been made to it then you will be prompted to save the changes. An Open file dialog will appear which you can use to navigate to the directory that contains the data file that you wish to edit. To select a file either double click on its name, or click on it once and click the '*Open/OK*' button. On opening the file the application will then check it for errors and if there are none it will be ready for use. If however there is an error an error message will be displayed.

Note: You only select the file with the *basefilename* for editing.

To save the data file once you have finished editing, use the '*Save*' or '*Save As*' options on the File menu. A dialog will appear when you select '*Save As*' to ask for a new filename. If you try and quit the application without saving when changes have been made to the table you will get a pop-up dialog asking you to either save the changes or discard them.

3.5.2   Select a table to edit

To select a table for editing, use the *Table* menu option. The contents of the drop down window change for each data file that is opened. A list of all the tables contained in the data file is given. To select a file simply click on the name of the table from the list. The currently selected table is marked on the drop down list.

3.5.3   View comments about a table

A set of comments may appear before each table in the data files, and this comment is associated with the table appearing after it. Before editing a file you might want to look at the comments that are located in the file containing the table. They may list any special information for the particular table. To view the comments select the '*View Comments*' option on the Table menu. **Note:** This option requires that a table is first selected. Where files contain multiple tables, the comments are associated with the table being viewed.

3.5.4   Selecting, Adding, and Removing Groups and variables within a group to a Table

To select which group to edit within a table you just click on one of the items contained in the list which initially has the title '*Group List*'. It is located in the top-left part of the screen. To add a group to the table you click the '*Add*' button, which is located under the groups list. A pop-up dialog will be shown with all of the valid group names which are available to insert into the table. To add one select it from the list and then click '*Ok*', or click '*Cancel*' to abort the add. Any newly added groups will appear at the bottom of the groups list. To remove a group from a table, select it from the groups list and then click the '*Remove*' button below the list. A confirmation dialog will appear before the group is actually removed.

Note: In version 1.0 of RadConEd if you have made changes to the GROUPS.DAT file you might need to exit and re-start the program before these changes become available for use.

All of the above mentioned operations are exactly the same for variables of the groups. When you click on a group in the groups list all of the variables for that group will appear in the variables list. Once they have appeared you can select, add or remove variables from the group.

3.5.5   Edit data for a Group/Variable combination

To edit the actual data values you first need to select a group, and then select a variable. You should now see numeric values in the data section (bottom half of the screen, with all of the cells). The data section has labels for the column names of the table you are editing and also for the section of the table you are editing: default value, Standard deviation (sd), minimum (min), maximum (max) or Distribution (dist). To edit a value click in the cell containing the value and then replace the current value with the new value. Entered values should be of either integer, decimal, or scientific notation($x.yyE^{\pm z}$). When you have finished editing values for a particular variable of a group click the '*Done*' button, or to cancel all changes since the last click of the '*Done*' button (or since the start of editing the variable values) click

the '*Cancel*' button. If you have changed values and then try and select another variable or group before pressing either the '*Done*' or '*Cancel*' button a pop-up dialog will appear asking if you want to keep the changes you have made. (This also occurs if you try and save the file or try and quit without either canceling any changes or updating the table with the '*Done*' button.)

3.5.6   Update/Undo the changes made

To update the table with changes you have made press the '*Done*' button. This will only update the table in the computers short term memory. Changes are not actually made permanent until the '*Save*' or '*Save As*' options on the File menu are used. To undo changes just made, press the '*Cancel*' button and the values before the last update was made will re-appear.

3.6   Using an existing scenario as the template
If you wish to use an existing scenario, i.e. other than that supplied by the distribution, as the template for a new scenario you need to follow the following preparation steps:
1. Create a '*zip*' file called '*basedatafiles.zip*' using the files in the scenario you wish to copy
2. Copy or move *basedatafiles.zip* to your current working directory
3. Create the new scenario using RadConEd.

3.7   Depositing the data in the RadCon directory
By default RadCon looks for the data files in a sub-directory *data/tables*, which it expects to exist in the directory from which it is being executed. To deposit the data files in the directory, if you were not editing these files, you will need to use system utilities to copy all the files from the directory where the required scenario is stored to the *data/tables* directory. It is recommended that the changes be made elsewhere and that the *tables* directory only be used when running RadCon.

Alternatively, RadCon can be instructed to use the data files directly from the directory where they have been stored.

3.8   Keeping a log of changes
RadConEd keeps a log of the changes that have been made to the data files in a file called CHANGES/LOG. To view any changes made to tables of a scenario you use the '*View Scenario Changes*' option from the File menu, select the scenario which you wish to see the changes from, by selecting the scenario name from the sub-menu which appears off of the '*View Scenario Changes*' option. A dialog will be shown that has a list of all changes made with details of the date, time, table, set, row and old and new values. This log can be exported (saved) to a comma delimited text file (useable for importing into Excel, Word etc) by clicking the '*Export*' button and selecting where to save the file and what to call it via a file save dialog.

3.9   Deleting a scenario.
To remove a scenario, open the File menu and select the scenario name from the sub-menu which will appear off to the right of the '*Delete Scenario*' option. A confirmation dialog will appear when a scenario has been selected to be deleted. Selecting yes to this confirmation will remove the scenario completely. USE WITH CAUTION !

3.10 Creating new tables in files

Version 1.0 of RadConEd does not allow the user to create new tables in a particular file, rather it only allows the editing of existing tables. This was not added due to time constraints. This feature may be added to RadConEd, as it could be needed in the future for the files containing the data for the diets of humans and animals.

To create new tables in RadCon the files may be edited using system editors..

4   Acknowledgements

The authors would like to acknowledge the RadCon team, F.F. Harris, J. Twining and D. Birtch for their contribution to this work.

5   References

[1] Crawford J., Domel R.U., Harris F.F., Twining J.R, May 2000: RadCon: A Radiological Consequences Model, Technical Guide, ANSTO E-744, ISBN 0-642-59982-3.

[2] Crawford J., Domel R.U. May 2000, RadCon: A Radiological Consequences Model, User Guide, ANSTO M-128, ISBN 0-642-59983-1.

[3] Domel R.U., Crawford J., Harris F.F., Twining J.R, March 2000: Parameter Report (Focusing on Tropical and Subtropical Environments in Australia). ANSTO Internal Report.

[4] Domel R.U., Crawford J., Harris F.F., Twining J.R, March 2000: Data, Assumption, Justifications and References used by the RadCon Model to Estimate Predictions for the Iput Test Scenario. ANSTO Internal Report.

A.   **Compiling the application**.

If you require to make any changes to the Java source files, you will need to re-compile the changed files.   Before compilation you need to ensure that the Java programming environment is installed on your system.   To compile the application use the command:

    javac TableEditor.java

If you get any errors please check that your CLASSPATH environment variable is set up and that it has the current directory as well as the JDK or JRE class files in it.

B.   **Basedatafiles.zip.**

This file contains the template data files and GROUPS.DAT file for scenarios to be created from. It is a standard '.zip' file and can be created with tools such as zip or gzip on UNIX® or for DOS/Windows users WinZip® or pkzip can be used to create the file. It is required that the GROUPS.DAT file contains all of the meta-data that defines the tables held in the files of the zip.

C.   Data files

The values for the parameters used by the RadCon program are grouped into 14 main files with the following names and content. The reader is referred to the RadCon Technical Guide [1] for further descriptions of the parameters:

- `AnimalInfo` - parameters required for the calculation of radionuclide biological excretion rates from contamination in animals. The parameters are:

Name in RadConEd	Parameter	Description	Units
Age	T	Age of animal at slaughter	years
Removal_Rate_1	$\lambda_{m1}(m,r)$	removal rate constant (fast) of radionuclide concentration in an animal due to physiological processes	d^{-1}
Removal_Rate_2	$\lambda_{m2}(m,r)$	removal rate constant (slow) of radionuclide concentration in an animal due to physiological processes	d^{-1}
Fraction_1	$\alpha_{m1}(m,r)$	fast removal component fraction	dimensionless

These are further described in [1] table 9 and equation 10.
- `Decay` - radionuclide decay constants in days, calculated as ln(2)/half-life
- `Deposition_Info` – parameters required for calculating the concentration in plants due to direct deposition and interception. These are described in tables 5,6 and 7 and are used in equations 5,6 and 7 in [1]. In addition the three parameters
  - ACT_LAI, Max_LAI and Yield depend on plant type only and not on the radionuclide, but for this version of RadCon they are reproduced for each radionuclide. If the variable Grass_k is non zero, RadCon will model that particular plant following grass, i.e. equation 6 in [1].

Name in RadConEd	Parameter	Description	Units
Dep_Vel	$V_{max}(r,p)$	maximum deposition velocity for plant p, i.e. for a fully developed plant	$m\ s^{-1}$
Act_LAI	LAI(p)	leaf area index of plant p at the time of deposition	$m^2\ m^{-2}$
Max_LAI	$LAI_{max}(p)$	maximum leaf area index of plant p at the time of fully developed foliage	$m^2\ m^{-2}$
Retention	S(p)	retention coefficient of plant p	mm
Yield	Y(p)	yield of plant p at time of harvest	$kg\ m^{-2}$
Yield	$Y_1(g)$	yield of grass at time of deposition	$kg\ m^{-2}$
Loss_rate	$\lambda_w(r,p)$	loss rate due to weathering of plant p or grass g	d^{-1}
Dilution_Rate	$\lambda_b(r,g)$	dilution rate by increase of biomass in grass	d^{-1}
Trans_Frac	$\alpha(r,g)$	fraction of activity translocated to the root zone in grass	dimensionless
Trans	$\lambda_t(r,g)$	rate of activity decrease due to translocation to the root zone in grass	d^{-1}
Grass_k	k	normalisation factor	m^2kg^{-1}

Note: Grass_k must be 1 for grass or 0 otherwise. Other values will give incorrect answers.
- `DoseCov` – contains the dose conversion factor for Inhalation, Ground_Shine, Cloud_Shine and Ingestion. The dose conversion factors for Inhalation and Ingestion (their units being Sv/Bq and depend on age and race of the individual) and those for Ground_Shine and Cloud_Shine (their units being (Sv m^2/(Bq s)).
- `HumanDiet` – contains the quantity (kg/day) of each food item consumed by infants, children, adult males, adult females and adults. **Note:** that valid data is required only for those age/sex groups that an assessment is to be carried out. In addition a vale for the processing field needs to be entered. This gives the fraction of radioactivity remaining in

the food after food processing i.e. before the food is consumed. See table 11 and equation 13 in [1].

The following three files are used in calculating dose from inhalation. For further information see the section on inhalation in [1].

- `InhalationRate` – contains the inhalation rate (for each of the age/sex groups (infant, child, adult male, adult female and adult). Further, different inhalation rates can be entered for the different races. **Note:** that the inhalation rate can be entered using this table or a combination of the next two tables, see equation 3 and table 3 in 1. If this option is selected the data in the following two tables has to be chosen such that

    I(race,age/sex_group) = Fraction of Time Resting x Resting Inhalation Rate +
      Fraction of Time Working x Working Inhalation Rate +
      Fraction of Time Alternate Activity x Alternate Activity Inhalation Rate = 1

    If the second alternative is chosen then the inhalation rate in this table must be set to 1 and the values in the next two tables must be chosen appropriately.

- `InhalationModifier` – the inhalation rate (m^3/hour) for each of the physical activities (working, resting and alternate) is entered here. In version 2.0 of RadCon these are race independent. This should be reconsidered for future releases of RadCon.

- `PhysicalActivity` – contains the fraction of time that each age/sex group undertakes each of the physical activities (working, resting and alternate).

Note: The data in the above two files is used to calculate the inhalation rate, I(race,age), using the above equation.

- `LocalConsumption` – contains the fraction of each food item consumed by humans that was produced locally. Any imported food is assumed to be un-contaminated. See table 11 and equation 13 in [1].

- `Occupancy` – contains the fraction of time that individuals spend indoors and outdoors. For version 2.0 of RadCon this is age and race independent. Considerations should be given to this in later versions of RadCon. These factors are used in dose calculations from inhalation and exposure from the cloud and ground. See the respective sections in [1] for further details.

- `SoilCharacteristics` – contains information on soil, see table 4 and equation 4 in [1].

Name in RadConEd	Parameter	Description	Units
Bulk_Density	P(s)	initial soil bulk density	kg m^{-2}
Migration_K	K(s)	soil reduction factor	dimensionless
Ground_Shine	Not used		

- `SoilRemoval-` contains soil information controlling the radionuclide availability for root uptake and contribution to external exposure. See tables 1 and 4 and equations 1 and 4 in [1].

Name in RadConEd	Parameter	Description	Units
The following relate to ingestion, table 4 and equation 4 in [1].			
Mobile_Fraction	$\alpha(r,s)$	mobile component fraction	dimensionless
Fixed_Component	$\lambda_1(r,s)$	rate of migration below root zone of mobile component of radionuclide	d^{-1}
Mobile_Component	$\lambda_2(r,s)$	rate of migration below root zone of fixed component of radionuclide	d^{-1}
Fixation	$\lambda_f(r,s)$	Rate of fixation	d^{-1}
The following relate to exposure from the ground, table 1 and equation 1 in [1].			
Gr_A	$\alpha(r,s)$	fixed component factor	dimensionless
Gr_Lambda1	$\lambda_1(r,s)$	migration rate, fixed component	d^{-1}
Gr_Lambda2	$\lambda_2(r,s)$	migration rate, mobile component	d^{-1}

- `Transfer` – contains a table with the soil to plant transfer rates (see equation 4 and table 4 in 1). The file also contains a second table with transfer rates from food consumed by animals to the animal products (see table 9 and equation 10 in [1]).

- `TransmissionFactors` – contains four tables with the shielding factors for indoors in case of cloudshine and groundshine and the filtering factors for inhalation. For further information on these see [1] tables 1,2 and 3 and equations 1, 2 and 3.

In addition for each animal product in the diet a file is required, animalDiet, which contains the daily consumption rates (kg/day) for each food in the animal's diet.

D.   Data file layout.

The data files have the following layout.

```
#comment1
#comment2
TITLE: tabletitle1
COLUMN_IDENTIFIERS:     columntitle1     columntitle2
UNITS:                  "column1 units"            "column2 units"
GROUPIDENTIFIER: Groupname1 (as defined in the GROUPS.DAT file)
VARIABLE IDENTIFER: Variablename1    column1value column2value
VARIABLE IDENTIFER: Variablename2    column1value column2value
VARIABLE IDENTIFER: Variablename3    column1value column2value
GROUPIDENTIFIER: Groupname2
VARIABLE IDENTIFER: Variablename1    column1value column2value
VARIABLE IDENTIFER: Variablename2    column1value column2value
VARIABLE IDENTIFER: Variablename3    column1value column2value

#comment1
#comment2
TITLE: tabletitle2
COLUMN_IDENTIFIERS:     columntitle1     columntitle2
```

The group identifiers and variable identifiers must be consistent within a single set of files (a set being the basefile and the other files with their prefixes). The units are required to be in quotation marks. None of the identifiers or group/variable names should contain white space. Also the column, group and variable identifiers must be present in the GROUPS.DAT file, and example of which has been given in Appendix E.   Ensure that a blank line is available between each table in the files.

The following shows the contents of `Transfer`. A value of –99.0 implies that the data for that field has not been set.

```
# SOIL to Crop transfer rates
# Depend on crop type and radionuclide
# There should be an entry in this table for every crop consumed by people and
TITLE: Transfer_Crop
```

ELEMENT:	Cs137	Sr-90	Xe133	I-131
UNITS:	"(Bq/kg)/(Bq/kg)"	"(Bq/kg)/(Bq/kg)"	"(Bq/kg)/(Bq/kg)"	"(Bq/kg)/(Bq/kg)"
SOIL: Sand				
FOOD_GROUP: Rice	0.128	-99.0	-99.0	-99.0
FOOD_GROUP: Grain	0.004	-99.0	-99.0	-99.0
FOOD_GROUP: Leafy_Vegetables	0.400	-99.0	-99.0	-99.0
FOOD_GROUP: Ground_Fruit	0.203	-99.0	-99.0	-99.0
FOOD_GROUP: Legumes_Pulses	0.045	-99.0	-99.0	-99.0
FOOD_GROUP: Root_Crops	0.096	-99.0	-99.0	-99.0
FOOD_GROUP: Tubers	0.21	-99.0	-99.0	-99.0
FOOD_GROUP: Tree_Fruit	0.041	-99.0	-99.0	-99.0
FOOD_GROUP: Poor_Grade_Feed	0.032	-99.0	-99.0	-99.0
FOOD_GROUP: High_Grade_Feed	0.21	-99.0	-99.0	-99.0
SOIL: Clay				
FOOD_GROUP: Rice	0.027	-99.0	-99.0	-99.0
FOOD_GROUP: Grain	0.014	-99.0	-99.0	-99.0
FOOD_GROUP: Leafy_Vegetables	1.099	-99.0	-99.0	-99.0
FOOD_GROUP: Ground_Fruit	0.807	-99.0	-99.0	-99.0
FOOD_GROUP: Legumes_Pulses	0.2	-99.0	-99.0	-99.0
FOOD_GROUP: Root_Crops	0.384	-99.0	-99.0	-99.0
FOOD_GROUP: Tubers	0.71	-99.0	-99.0	-99.0
FOOD_GROUP: Tree_Fruit	0.014	-99.0	-99.0	-99.0
FOOD_GROUP: Poor_Grade_Feed	0.022	-99.0	-99.0	-99.0
FOOD_GROUP: High_Grade_Feed	0.027	-99.0	-99.0	-99.0
SOIL: Loam				
FOOD_GROUP: Rice	0.015	-99.0	-99.0	-99.0
FOOD_GROUP: Grain	0.0042	-99.0	-99.0	-99.0
FOOD_GROUP: Leafy_Vegetables	0.217	-99.0	-99.0	-99.0
FOOD_GROUP: Ground_Fruit	0.015	-99.0	-99.0	-99.0
FOOD_GROUP: Legumes_Pulses	0.173	-99.0	-99.0	-99.0
FOOD_GROUP: Root_Crops	0.501	-99.0	-99.0	-99.0
FOOD_GROUP: Tubers	0.009	-99.0	-99.0	-99.0
FOOD_GROUP: Tree_Fruit	0.263	-99.0	-99.0	-99.0
FOOD_GROUP: Poor_Grade_Feed	0.0136	-99.0	-99.0	-99.0
FOOD_GROUP: High_Grade_Feed	0.028	-99.0	-99.0	-99.0
SOIL: Coral				
FOOD_GROUP: Rice	-99.0	-99.0	-99.0	-99.0
FOOD_GROUP: Grain	4.439	-99.0	-99.0	-99.0
FOOD_GROUP: Leafy_Vegetables	40.9	-99.0	-99.0	-99.0
FOOD_GROUP: Ground_Fruit	3.89	-99.0	-99.0	-99.0
FOOD_GROUP: Legumes_Pulses	3.90	-99.0	-99.0	-99.0
FOOD_GROUP: Root_Crops	-99.0	-99.0	-99.0	-99.0
FOOD_GROUP: Tubers	3	-99.0	-99.0	-99.0
FOOD_GROUP: Tree_Fruit	8.37	-99.0	-99.0	-99.0
FOOD_GROUP: Poor_Grade_Feed	7.41	-99.0	-99.0	-99.0
FOOD_GROUP: High_Grade_Feed	8.23	-99.0	-99.0	-99.0

```
#
# Plant to Animal transfer rates
# Depend on Animal type and radionuclide
# There should be an entry in this table for all meat types consumed by
# people
TITLE: Transfer_Animal
```

ELEMENT:	Cs137	Sr-90	Xe133	I-131
UNITS:	"(Bq/kg)/(Bq/kg)"	"(Bq/kg)/(Bq/kg)"	"(Bq/kg)/(Bq/kg)"	"(Bq/kg)/(Bq/kg)"
SOIL: Animal				
FOOD_GROUP: Cow_Milk	0.0079	0.0028	0	0.01
FOOD_GROUP: Cow	0.05	0.008	0	0.04
FOOD_GROUP: Pig	0.24	0.04	0	0.0033
FOOD_GROUP: Eggs	0.4	0.0028	0	0.01
FOOD_GROUP: Fowl	3.0	0.08	0	0.01
FOOD_GROUP: Sheep	0.49	0.33	0	0.03

The data in the *minTransfer, maxTransfer, sdTransfer* and *distTransfer* files will be given in the same order and will have the same format. Although the TABLE names will have a prefix of *min, max, sd* and *dist* respectively. e.g. *distTransferCrop, distTransfer_Animal.*

E.   GROUPS.DAT meta-data file

The GROUPS.DAT metadata file has the following layout:

GROUPNAME: Variable1   Variable2   ….   Variable'N'
Files: AssociatedFile1.dat=Constraint(eg ALL or ANY)
                AssociatedFile2.dat=ANY
                …
                AssociatedFile'N'.dat=ALL

The file requires that a blank line be left between separate groups defined in the file.
An example is as follows:

```
ELEMENT:        Cs137  Sr-90  Xe133  I-131
Files: Decay=ALL
            DoseConv=ANY
            Deposition_Info.dat=ALL
            SoilCharacteristics=ALL
            SoilRemoval=ALL
            Transfer=ALL
            TransmissionFactors=ANY

SOIL:  Sand   Loam   Clay   Coral
Files: AnimalDiet=ALL
            SoilInfo=ANY

RACE:  Caucasian     Asian   Indian  Islander
Files: DoseConv=ALL
            HumanDiet=ALL
            InhalationRate=ALL

ACTIVITY:       Working Resting Alternate
Files: InhalationRate=ALL
            LifeStyle=ALL

LOCATION:       Rural  Residential    Urban
Files: Transmission=ALL
            LifeStyle=ALL

LIFE_STYLE:     Indoor Outdoor
Files: Transmission.factor=ALL
            LifeStyle.dat=ALL

AGE_GROUP:      Infant Child  Adult_Male     Adult_Female   Adult
Files: HumanDiet =ALL
            InhalationRate=ALL

SOIL_VARIABLES_1:    Bulk_Density  Migration     Ground_Shine
Files:      SoilCharacteristics=ALL
            SoilRemoval=ALL

SOIL_VARIABLES_2:    Mobile_Fraction    Fixed_Component    Mobile_Component
        Fixation     Gr_A   Gr_Lambda1   Gr_Lambda
Files: SoilCharacteristics=ALL
            SoilRemoval=ALL


DEPOSITION_VARIABLES:        Act_LAI Max_LAI Yield Dep_Vel  Retention     Loss_rate
        Dilution_rate Trans_Frac   Trans  Grass_k
```
Note: this is a wrap around from the previous line, ie all the variables should be on one line
```
Files: DepositionInfo=ALL

ANIMAL_VARIABLES: Age Fraction_1     Removal_Rate_1 Removal_Rate_1
Files: AnimalInfo=ALL
FOOD_GROUP:     Grain  Leafy_Vegetables    Vegetables    Root_Vegetables      Fruit
        Fodder Potatoes      Berries_Wild  Mushrooms_Wild Maize  Straw  Hay    Legumes
        Ground_Fruit  Tubers High_Grade_Feed     Poor_Grade_Feed    Sumpplement
        Vegetation    Pig  Fowl  Cow   Cow_Milk     Eggs
```
Note: once again all the variables are on the same line
```
Files: AnimalDiet=ALL
            HumanDiet=ALL
            Transfer=ALL
            Deposition_Info= ALL
```

F. A Possible Procedure for editing files

The following procedure could be used in editing your files in preparation for use by RadCon:

1. Create a directory where the distribution of RadConEd will be stored.
2. Follow the installation procedure to install the RadConEd system on your machine, see section 2.1.
3. Select a working directory, see section 2.2 to prepare for running the system.
4. Copy *basedatafile.zip* in the working directory.
5. Start the RadConEd system and create your new scenario.
6. Edit Scenario Groups to add any new variable names etc. see section 3.5.
7. Edit the tables, see section 3.5.
8. Copy the new tables to the RadCon working directory, under subdirectory *data/tables*, or when using RadCon instruct it to use the actual location of the data files.