

SCIENTIFIC SOFTWARE & MODELLING SOLUTIONS

AMBER 4.5 Reference Guide

Version 1.02e

July 2003

Enviros Consulting Limited
Building D5 • Culham Science Centre
Culham • Oxfordshire
OX14 3DB
United Kingdom
Phone (44) 1235 468800
Fax (44) 1235 468828
email amber@enviros.com
www.enviros.com/amber/

Quintessa Limited
Dalton House • Newtown Road
Henley-on-Thames • Oxfordshire
RG9 1HG
United Kingdom
Phone (44) 1491 636246
Fax (44) 1491 636247
email amber@quintessa.org
www.quintessa.org/amber/

© Enviros Consulting Limited. All rights reserved.

Microsoft and MS-DOS are registered trademarks of Microsoft Corporation.
Windows and Windows NT are trademarks of Microsoft Corporation.

Important Note

This document should be read in conjunction with the 'Release Note for Version 4.5', and the AMBER 'Getting Started' Guide

AMBER 4.5 Reference Guide



PREFACE

AMBER is a flexible software tool that allows the user to build their own dynamic compartmental models to represent the migration and fate of contaminants in a system, for example in the surface and sub-surface environment. Contaminants in solid, liquid and gaseous phases can be considered. AMBER gives the user the flexibility to define:

- any number of compartments;
- any number of contaminants and associated decays;
- any number of transfers between compartments;
- algebraic expressions to represent physical processes operating between compartments;
- algebraic expressions to represent the uptake of contaminants by humans, other biota and additional output quantities of interest;
- deterministic, probabilistic and time varying parameter values.

This flexibility means that users can construct case-specific models and tailor generic models to their specific needs.

Key features incorporated into AMBER include:

- powerful, user-friendly graphical interface which gives AMBER the "look and feel" of a Windows application;
- fast and accurate Laplace transform and numerical time-step solver;
- time varying source terms and transfer processes;
- ingrowth and decay/degradation of contaminants;
- built in graphing of results;
- tailoring of results format.

AMBER 4.5 Reference Guide

AMBER is being developed by EnviroS and Quintessa. Additional technical and/or financial assistance has been provided by Centro de Investigaciones Energéticas Medioambientales y Tecnológicas (CIEMAT), Universidad Politécnica de Madrid (UPM), Empresa Nacional de Residuos Radiactivos S A (ENRESA), JGC Corporation, the Japan Nuclear Cycle Development Institute (JNC), the Swedish Nuclear Power Inspectorate (SKI), the United Kingdom Atomic Energy Authority (UKAEA) and Nirex.

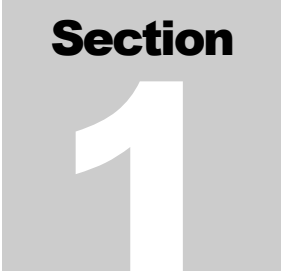
The user documentation for AMBER 4.5 consists of three volumes: 'Getting Started', the 'Reference Guide' and the 'Release Note for Version 4.5'. In 'Getting Started', the user is introduced to AMBER through tutorial exercises. The tutorial demonstrates the functionality of AMBER during the construction of a relatively simple compartment model. It also allows the user to gain experience in setting up a model within AMBER. The Reference Guide describes system requirements and installation and provides further detailed information on the use of AMBER, plus a glossary of AMBER related terms. The 'Release Note' provides detailed information regarding the latest version of AMBER, that is Version 4.5.

Table of Contents

1. INTRODUCTION	5
1.1 OVERVIEW OF AMBER	5
1.2 THIS GUIDE	7
2. INSTALLING AMBER.....	9
2.1 SYSTEM REQUIREMENTS	9
2.2 INSTALLATION PROCESS	9
2.3 BACKWARD COMPATIBILITY.....	12
3. STARTING AMBER.....	13
3.1 START-UP WINDOW	14
3.2 MENU BAR	14
3.3 THE TOOLBAR	19
4. CONVENTIONS IN AMBER	20
4.1 NAMING.....	20
4.2 NUMERICAL FORMATS.....	21
4.3 UNITS	21
4.4 DESCRIPTIONS	31
5. SETTING UP A CASE FILE.....	33
5.1 MODEL WINDOW	33
5.2 CONTAMINANTS	42
5.3 DECAYS.....	45
5.4 PARAMETERS.....	51
5.5 EXPRESSIONS.....	84
5.6 MULTIPLICITY.....	91
5.7 RESULT TIMES.....	92
5.8 SAVING THE CASE.....	96

AMBER 4.5 Reference Guide

6. SAMPLING.....	99
6.1 UNIFORM.....	101
6.2 GAUSSIAN.....	102
6.3 TRUNCATED GAUSSIAN.....	102
6.4 TRIANGULAR.....	102
6.5 BETA.....	103
6.6 GENERAL PROBABILITY DISTRIBUTION FUNCTIONS.....	103
6.7 PARTIAL SAMPLING.....	104
6.8 SEEDS.....	105
6.9 SELECTING SPECIFIC SAMPLES.....	105
7. RUNNING AND REPORTING A CASE FILE.....	107
7.1 CHECKING PARAMETERS.....	107
7.2 CALCULATION.....	109
7.3 RESULTS.....	114
8. ADDITIONAL FEATURES.....	127
8.1 MODEL WINDOW BASICS.....	127
8.2 SUBMODEL HIERARCHIES.....	133
8.3 AVAILABILITY SCHEMES.....	138
8.4 PRECONDITIONS.....	144
8.5 COMMAND LINE ARGUMENTS.....	146
9. TROUBLE SHOOTING AND FEEDBACK.....	149
9.1 ON-LINE HELP.....	149
9.2 ERRORS AT STARTUP.....	149
9.3 PARAMETERS.....	150
9.4 ERROR MESSAGES.....	151
9.5 COMMON SYNTAX ERRORS.....	153
9.6 COPY AND PASTE.....	153
9.7 CHOOSING RESULT TIMES.....	154
9.8 SPEEDING UP THE TIME-STEP SOLVER.....	154
9.9 FEEDBACK.....	154
10. RESTRICTED VERSIONS OF AMBER.....	155
APPENDIX A. GLOSSARY OF TERMS.....	159

A graphic consisting of a grey square with the word "Section" in bold black text at the top and a large white number "1" in the center.

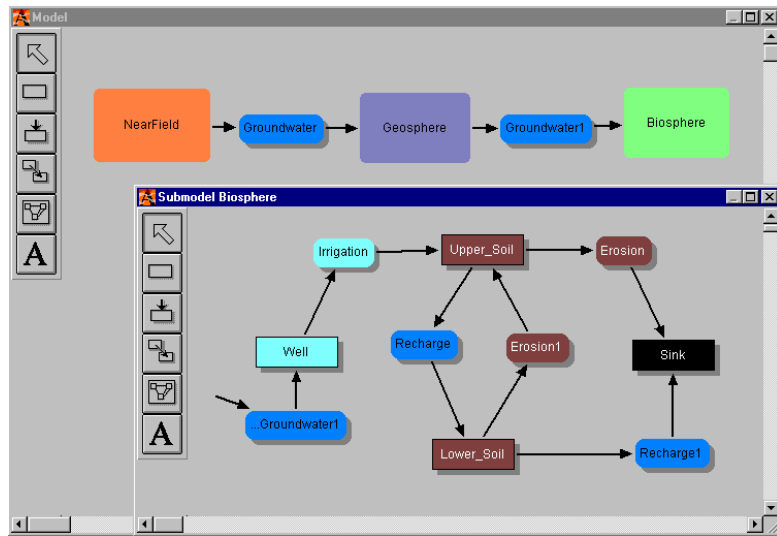
1. Introduction

1.1 Overview of AMBER

In AMBER, the materials of interest, referred to as contaminants, are assumed to be present in a series of compartments between which transfers can take place. A compartment is any specific part of the system being modelled, for instance a lung in a model of the body. Within any given compartment, a contaminant is assumed to be uniformly mixed. Each transfer is ‘donor controlled’, depending directly on the amount of the material present in the compartment from which the material is moving, and can change with time. AMBER allows contaminants to decay or degrade with time into other contaminants, if required. For example, one radionuclide may decay into another, or a particular chemical may be degraded into a different one.

A compartment model usually contains compartments, transfers between compartments and sources providing input of contaminants to compartments (see, for example, *Figure 1.1-1*).

Figure 1.1-1
An Example
Compartmental
Model



Mathematically, the amount of contaminant in any compartment is determined by the equation (1).

If the total amount of contaminant m in compartment i is I_i^m (moles) then this satisfies:

$$\frac{dI_i^m}{dt} = - \left[\lambda_r^m + \sum_j \lambda_{ij} \right] I_i^m + \lambda_r^{m+1} I_i^{m+1} + \sum_j \lambda_{ji} I_j^m \quad (1)$$

where λ_{ij} is the exchange rate between compartment i and compartment j (y^{-1}), λ_r^{m+1} is the decay rate of the parent contaminant $m+1$ (y^{-1}), and λ_r^m is the decay rate of contaminant m (y^{-1}).

These equations are linear, so that, for example, if there is twice as much contaminant in the system initially, then the calculated

concentrations in the various compartments will all be doubled at each time of interest.

Although the models which can be set up in AMBER are generally limited to those of the linear donor controlled compartment type, this class of models can be applied to a very wide range of problems as a result of the flexibility with which transfers between compartments can be specified. For example, some diffusive-like processes depend upon the concentrations of contaminants in both the donor and receiving compartments, but these can readily be represented in AMBER by including a 'forward' transfer from the donor to the receiving compartment and a 'backward' transfer from the receiving compartment to the donor. The combination of these two transfers will correctly model the net transfer between compartments. Certain non-linear problems, such as those arising from solubility limitations, can also be solved using AMBER.

AMBER has been applied to a wide range of problems concerned with the way that radionuclides and other contaminants move through different parts of the environment.

1.2 This Guide

This is a guide to the use of AMBER 4.5 for Windows 95/98/NT/2000/ME and XP. This guide provides instruction on the installation and operation of AMBER.

Section 2 provides details regarding the installation of AMBER 4.5. Section 3 guides the user around the basic features of the GUI and getting started. Section 4 details certain formatting and unit conventions in AMBER. Sections 5 and 6 explain creating, running and reporting case files and Section 7 details some recent features of AMBER. A glossary of terms is given in Appendix A.

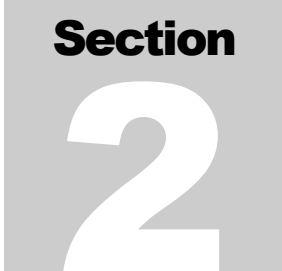
AMBER makes use of many standard Windows features including:

- ▲ Scroll bars - to facilitate moving around the Model window and the description fields;
- ▲ Cut, copy and paste functions - to simplify the entry of complicated expressions ;
- ▲ Pull down menus - offering extensive functionality but a tidy interface;
- ▲ Toolbar - icons providing a quick route to frequently used functions;
- ▲ Check boxes and radio buttons - to select options

and many more. Users who are unfamiliar with a Windows environment should consult their system documentation for information on how to use these features.

Throughout this guide 'click' means clicking the left button of the mouse and 'double click' means clicking the left button of the mouse twice in quick succession. Where the right button of the mouse needs to be clicked, this guide uses 'right click'.

Menu items are denoted in the text by *italics*. The '|' notation is used in this guide for menu separation. For example *Results | Calculate* means use the mouse to select the *Results* menu from the AMBER menu bar, drag down to the menu item marked *Calculate* and release the mouse button. <return> means press the 'return' / 'enter' key. Commands and text that can be typed are shown in `courier` font.

A graphic consisting of a grey square with the word "Section" in bold black text at the top and a large white number "2" in the center.

2. Installing AMBER

2.1 System Requirements

The recommended system requirements needed to run AMBER are a PC with a Pentium processor with at least 16 Mb of RAM installed, running under the Windows '95, '98, 2000, ME, XP or the Windows NT (version 4.0) operating system. AMBER will run on lower specification machines but its performance will be reduced, e.g. calculations will run more slowly. At least 10 Mb of hard disk space should be available. The PC screen resolution used should be set to at least 800 × 600 pixels and a small font should be used.

2.2 Installation Process

AMBER 4.5 is installed from one compact disc. Place the CD in the CD drive. Select *Start | Run*. Type [CD drive]:\setup.exe, e.g. f:\setup.exe. Click on 'OK'. Follow the installation instructions. A Program Manager/Start group and icons will be created automatically. An uninstaller will

also be installed which you can use to remove AMBER from your system.

The following files will be installed:

- ▲ AMBER.exe (the executable);
- ▲ *.dll (dynamic link libraries);
- ▲ *.irs (text resources);
- ▲ *.cse (example AMBER case files, see File | Case Information for more on each case.);
- ▲ QCharter.exe (for graphing);
- ▲ readme.txt (information about this version of AMBER).

If you are installing a fully functional copy of AMBER (not a demonstration version) you will need to plug the dongle, provided by Enviros, into the printer (parallel) port before you can run it. No dongle is required for the demonstration version.

If you are using Windows NT, Windows 2000 or Windows XP it is necessary to install a dongle device driver. Follow the instructions supplied with AMBER or consult the FAQ section of the AMBER website to install the driver.

With the release of AMBER Version 4.5 we have supplied an updated dongle device driver (Version 5.41.1). However, **if you have previously installed a suitable device driver** (e.g. from a previous install of AMBER) you generally will **not** need to re-install the updated version of the driver.

Quick installation guide:

1. Log on with Administrator privileges (if applicable).
2. Insert the AMBER 4.5 Install Disk into your CD-ROM drive.
3. From Windows:
 Either:
 In Explorer, select the appropriate CD-ROM drive and double-click 'Dongle_Install_Win2k_NT_XP(32bit).exe'. Alternatively, if you are running a 64-bit version of XP double-click on 'Dongle_Install_XP(64bit).exe'.
 or:
 From your desktop click on 'Start' then click on 'Run' Type '<Drive letter>:\Dongle_Install_Win2k_NT_XP(32bit).exe' at the prompt. Alternatively, if you are running a 64-bit version of XP then type '<Drive letter>:\Dongle_Install_XP(64bit).exe'. Press the 'OK' button.
4. Follow the typical installation prompts, until installation is complete.
5. It is recommended that you 'Restart' your computer before using AMBER.
6. Once installed the driver software can, if necessary, be removed/repaired/stopped by running the file 'SetupSysDriver.exe' located by default in the following directory: 'C:\Program Files\Rainbow Technologies\Sentinel System Driver'.

Additional information on the dongle device drivers can currently be found at:

http://www.rainbow.com/support/eu_support.htm;

or after installation of the driver software, in the 'Rainbow Technologies' directory on your computer (see above).

No device driver is need for Windows 95/98 or ME.

2.3 Backward Compatibility

AMBER 4.5 is fully backward compatible with case files from AMBER 2.0, 3.0, 3.1, 3.3, 4.0, 4.1, 4.2, 4.3 and 4.4. Case files produced in AMBER 4.5 contain additional information and are not readable in previous versions of AMBER.

Section**3**

3. Starting AMBER

To start the application click on the AMBER icon in the Programs section of the Windows start menu.

If you wish to start AMBER in a different directory to the '.irs' files you will have to edit the 'RESOURCE_PATH' environment variable. Consult your system manual for how to set this environment variable. When using NT 4.0, the 'RESOURCE_PATH' can be read or modified through the 'Start' menu option 'Control panel'. Double click on 'system' and then select 'RESOURCE_PATH' from the list box of 'User variables'. For example, you may wish to change the value of this variable to:

```
c:\AMBER_4
```

If you do set this variable remember you will have to change it if you install a new version of AMBER.

3.1 Start-up Window

When you first start AMBER a Start-up window will appear. This window will disappear after a few seconds. If you do not want to wait you can clear it straight away by clicking on it.

3.2 Menu Bar

After the Start-up window has cleared the menu bar will be visible. The menu bar contains *File*, *Edit*, *Results*, *Options*, *View*, *Windows*, *Font* and *Help* menus. Note that items in some of these menus may be 'greyed out'. This means that the option is not available at the current stage of the calculation.

Keyboard shortcuts can be used to speed up working. For example (Ctrl + N) means hold down the 'Ctrl' key and press the 'N' key. These are noted in the following sections.

3.2.1 File Menu

The following items are available from this menu:

- ▲ New Case (Ctrl+N) : Opens a new case;
- ▲ Open Case (Ctrl+O) : Opens an existing case;
- ▲ Close Case : Closes the current case;
- ▲ Save Case (Ctrl+S) : Saves the current case and makes a backup file of the previously saved case (with the extension '.cbk');
- ▲ Save Case As : Saves the current case with a new file name;

- ▲ Save Special: This allows a case file to be saved that only contains a selected set of contaminants (identified by a Selection). Any decays from selected contaminants to those that are excluded are reassigned to be from the selected contaminant to NULL. After the Save Special, the reduced contaminant case file is in memory. The option is only available when the original case file does not need saving (to avoid loss of data).
- ▲ Revert to Saved : Reverts to the previously saved case;
- ▲ Print Setup : Sets up the printer options;
- ▲ Print : Prints the active window (Model window only);
- ▲ Exit : Quits the application;
- ▲ Case Information: Displays information on the current case. The case name and case description can be viewed. Note that the case name may be edited, and need not be the same as the case file name;
- ▲ A list of currently open cases: The names of all currently open cases are displayed and the case currently being worked on is marked with a tick. Any of the background cases can be brought to the front by selecting the required case from the list.

There is a maximum of five open cases at any one time.

3.2.2 Edit Menu

The only item available from this menu is:

- ▲ Delete (Ctrl+Del): Deletes selected compartments, transfers and sources from the Model window. (N.B. The Cut, Copy, Paste items are currently always disabled.)

3.2.3 Results Menu

The following items are available from this menu:

- ▲ Calculate (F9) : Begins the calculation;
- ▲ Check Parameters : Produces a detailed report of all the user defined parameters and of solver inputs such as Transfer rates (Note that a report on observer parameters is only available once a calculation has been completed);
- ▲ Check Preconditions: This lists failures, and flags errors and warnings. The user-specified failure message is written out for each failure
- ▲ Graph (Ctrl + G) : Creates a graph;
- ▲ Report (Ctrl + R) : Outputs a report to a text file;
- ▲ Sensitivity: Outputs sensitivity information to a text file;
- ▲ Dump: Outputs a detailed summary of a calculation to a text file.

3.2.4 Options Menu

The items available from this menu are:

- ▲ Confirm deletes: Allows the user to toggle on/off whether they should be prompted for confirmation before deleting an item;

- ▲ Units: Allows the user to set the units of ‘time’ and ‘amount’ from a variety of choices, and to specify how AMBER should treat unit errors, see *Section 4.3*;
- ▲ Locks: Parameters may be locked as described in *Section 5.4.6*.

3.2.5 View Menu

The View menu contains sub-menus for:

- ▲ selection and de-selection;
- ▲ zooming;
- ▲ colour;
- ▲ relative size; and
- ▲ stored layout management.

3.2.6 Windows Menu

The following items are available from this menu: Please refer to the glossary for definitions.

- ▲ Model;
- ▲ Submodels;
- ▲ Contaminants;
- ▲ Decays;
- ▲ Compartments;

- ▲ Transfers;
- ▲ Sources;
- ▲ Parameters;
- ▲ NameSets;
- ▲ Mappings;
- ▲ Result Times;
- ▲ Switch Parameters;
- ▲ SnapShot Parameters;
- ▲ Pre-conditions.

Selecting any of these opens the corresponding window for the case. If the window is already open it will be moved to the front.

3.2.7 Font

Allows the user to choose the font for the text which appears in the Model window from those currently installed on the PC.

Note that the font used when zooming the Model window will be a scaled version of that selected.

3.2.8 Help Menu

The only items currently available from this menu are:

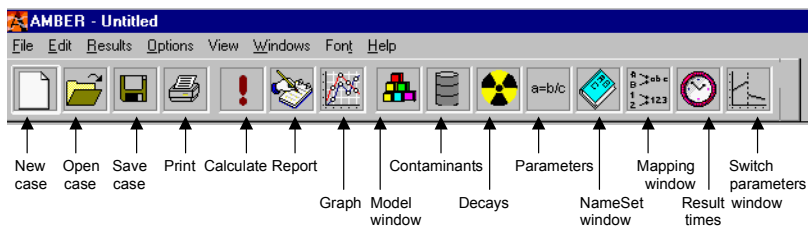
- ▲ About AMBER: gives information about AMBER;

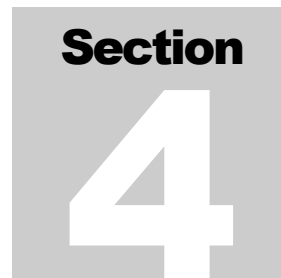
- ▲ Reference guide : provides an on-line version of this document.

3.3 The Toolbar

AMBER 4.5 has an icon toolbar (see *Figure 3.3-1*) for the most commonly used commands from the menu bar. The icons have ‘toolbar tips’ associated with them, which appear when the cursor moves over the icon, and acts as a reminder of the function. The ‘tips’ will disappear after a second or so.

Figure 3.3-1
The Toolbar



A gray square containing the word "Section" in a bold, black, sans-serif font at the top, and a large, white, bold, sans-serif number "4" in the center.

4. Conventions in AMBER

4.1 Naming

The names of parameters, NameSets, literals, contaminants, compartments, sources and transfers are subject to certain restrictions. A name must start with an alphabetic character, followed by a sequence of characters that are alphanumeric or ‘_’ (underscores). **Space characters and ‘-’ (minus) characters are not allowed in names.**

No two of the following can have the same name: contaminant, NameSets, literals, compartment, source and transfer. A parameter can have the same name as a contaminant, compartment, source or transfer, but this is not recommended. The name ‘NULL’ has special significance and cannot be used. Names in use as internal parameters, such as ‘Amount’, cannot be used for user defined parameters.

Case is significant, for example ‘river’, ‘River’ and ‘RIVER’ are all considered to be different names.

4.2 Numerical Formats

All of the following numerical formats are acceptable as constants by AMBER:

0.001
1e-3
1E-3
1.0E-3

4.3 Units

4.3.1 User-Defined Units

A feature first incorporated into AMBER 4.4 is that AMBER is now aware of units and can manipulate them to convert from one unit to another. All expressions are checked for unit consistency.

Each parameter has a defined unit, either set by AMBER or by the user. The unit is entered through the Parameter Edit dialog box, in a new field just below the name, by selecting *Options | Units...* from the menu (see *Figure 4.3-1*). The format of this is discussed later. For every expression, and every operation within an expression, units consistency is checked and conversion factors used if required. All outputs (graphs and reports) include the units of the parameter.

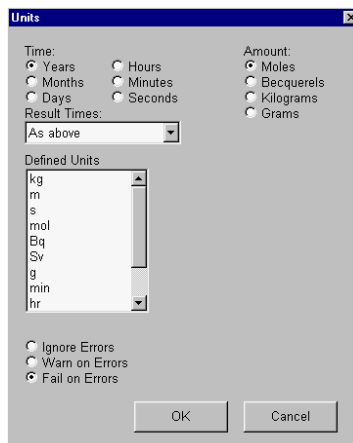
Note that all expressions must have units that are consistent with the parameter. If the expression is simply a constant then AMBER assumes that it has the units of the parameter and so the units need not be explicitly included in the expression.

The units for internal AMBER quantities are selected through the Units Dialog. These relate to sources, initial amounts, transfers, decays and times. For example, if the user selects 'Seconds' and

'Kilograms' all source fluxes will be interpreted as being in units of 'Kilograms per Second' and all decay rates will be interpreted as being in units of seconds. Changing the units after setting up a model should be cautioned against since data values already entered may need to be updated to account for the change in units. Thus if a contaminant had a decay rate of 0.1 per year, if the units of time were changed to seconds, the decay rate would be assumed to be 0.1 per second.

Figure 4.3-1

The Units Dialog



Note that this dialog also lists the available units (see *Section 4.3.1.1*) and includes radio buttons for controlling how AMBER treats unit errors, see *Section 4.3.3*.

Decays are calculated in terms of moles. For example, 1 mole of A decays to 1 mole of B. However, this one-to-one mapping is not necessarily true of Kilograms, Grams or Becquerels. Consequently, if the user selects the units of amount to be Kilograms or Grams, an atomic mass which is greater than zero must be supplied for each contaminant so that decay calculations can be carried out. Similarly, if the user selects the amount units to be Becquerels, each

contaminant must have a non-zero decay rate so that decay calculations can be performed.

The default setting for time units is 'Years' and 'Moles' for amount units. A case file from a version of AMBER before AMBER 4.0 will be assumed to be in these units.

The user can select different result time units from the ones used within the model. This will only affect the times reported, for further information see *Section 5.7.1*, Result time units.

4.3.1.1 Units Available

AMBER has the following units built in, and can handle prefixed versions indicated (not all of which are necessarily useful).

Table 4.3-1
AMBER Units
and Prefixed
Units

Unit	Definition	Prefixed units
kg	basic unit	
m	basic unit	nm, um*, mm, km, Mm, Gm, Tm
s	basic unit	ns, us, ms, ks, Ms, Gs, Ts
mol	basic unit	
Bq	basic unit	nBq, uBq*, mBq, kBq, MBq, GBq, TBq
Sv	basic unit	nSv, uSv*, mSv, kSv, MSv, GSv, TSv
g	0.001 kg	ng, ug*, mg, Mg, Gg, Tg
min	60 s	
hr	3600 s	
day	24 hr	
y	365.25 days	
month	1/12 y	

* the prefix u is used to represent micro because of its similarity to the Greek μ .

It is not possible to extend this list through the GUI, but the case file can be edited to add new units. If a case file is saved in Version 4.5 there will be a block EXTRA_UNITS that can be added to.

To add a new basic unit, add a line BASE followed by an optional line PREFIXABLE and then a line with the unit name, e.g.

```
BASE
PREFIXABLE
V
```

would add the units V (volts) and allow mV, kV etc.

To add a derived unit, the first line is DERIVED, optionally followed by PREFIXABLE then the unit name and finally a line giving the definition. For example

```
DERIVED
PREFIXABLE
N
1 kg m s-2
```

add N (Newton), while

```
DERIVED
cm
0.01m
DERIVED
inch
2.54 cm
```

adds both cm and inch.

All derived units must be linked back to basic units through their definitions.

The full list of available units is given in the GUI under the Units Dialog.

4.3.1.2 Units Syntax

The syntax for specifying units is quite flexible.

A dimensionless unit can be indicated by a null entry or by a single dash (-). Otherwise units are written with separating spaces and powers directly attached to the relevant component. The following are all valid.

m
m²
m^{^2}

m⁻²
m²/s
m² s⁻¹
m^{^2} s^{^-1}

It is also valid (although not recommended) for the unit to be preceded by a number, so an item with units of km could have this specified as 1000m.

When a unit is attached to a constant in an expression, it is contained in square brackets, e.g. $x+1[m]$ is a valid expression (if x has consistent units).

4.3.2 Unit Conversion Parameters

To convert between quantities in different units, internal conversion parameters are provided.

The internal parameters for unit conversion are given in *Table 4.3-1*.

'MoleToKg' and 'KgToMole' can only be used for contaminants with atomic masses >0, for example if contaminant A has atomic mass 100 and contaminant B has atomic mass 0 a 'MoleToKg[A]' can be used, but not 'MoleToKg' or 'MoleToKg[B]'. Similarly 'MoleToBq' and 'BqToMole' can only be used for contaminants with a decay rate >0.

Table 4.3-2
Internal Unit
Conversion
Parameters

Parameter	Multiplicity	Value
BqToMole	contaminant	1/MoleToBq
KgToMole	contaminant	1/MoleToKg
MoleToBq	contaminant	$6.022 \times 10^{23} *$ DecayConstant * YearToSec
MoleToKg	contaminant	AtomicMass/1000

The conversion factors for different time units that were provided in versions of AMBER, before 4.4, have been redefined to use the units conversion capability. For example YearToDay is equal to 1 [y/day].

4.3.2.1 Units Consistency and Conversion

AMBER checks unit consistency in each operation and each function. The rules are listed below in *Table 4.3-3*. Convertible units are those that differ only by a constant factor.

Table 4.3-3
Unit
Consistency
Rules

Operation/Function	Consistency Rule	Result Units
A+B, A-B	A and B convertible	as A
A*B, A/B	none	Product/quotient of units
A^B	B dimensionless (integer unless A dimensionless)	Power of units of A
A<B, A<=B, A>B, A>=B, A==B, A!=B	A and B convertible	Dimensionless
exp(A), ln(A), log10(A), sin(A), cos(A)	A dimensionless	Dimensionless
abs(A), int(A)	None	as A
sqrt(A)	A must have square units	square root of units of A
_not(A)	A dimensionless	dimensionless
or(A,B...), and(A,B...), xor(A,B...)	all dimensionless	dimensionless
min(A,B...), max(A,B...)	all convertible to A	as A
stepUp(A,B), stepDown(A,B)	A and B convertible	dimensionless
TopHat(A,B,C)	A, B and C convertible	dimensionless
sum(A), sum(A, nameset), largest(A), largest(A, nameset)	None	as A
sumType(A, mapping)	None	as A

Table 4.3-3
Unit
Consistency
Rules

Operation/Function	Consistency Rule	Result Units
typeLookUp(A, mapping)	None	as A
_LOOKUP(X, Y0, X1, Y1...)	X's convertible or dimensionless, Y's convertible	as Y1
_CYCLIC_LOOKUP(X, Y0, X1, Y1...)	X's convertible or dimensionless, Y's convertible	as Y1
_LINEAR_LOOKUP(X, X1, Y1...)	X's convertible or dimensionless, Y's convertible	as Y1
lookUp (X, Y0, X1, Y1...)	X's convertible, Y's convertible	as Y1
cyclicLookUp(X, Y0, X1, Y1...)	X's convertible, Y's convertible	as Y1
linearLookUp(X, X1, Y1...)	X's convertible, Y's convertible	as Y1

4.3.3 Units Warnings and Errors

Units checking can be set through the Units Dialog to generate a warning rather than errors, so that old cases (cases created in versions of AMBER earlier than version 4.4) can be handled. It is recommended that units are added to these cases and that the units checking is set to generate errors once no warnings are being given.

The checking can also be set to ignore errors, in which case no warnings are given. When the unit errors are ignored or treated as warnings, AMBER just uses the numeric values.

Switching between warnings and errors is done through a check box on the Units dialog.

The warning messages given will generally be about consistency between units. This can happen when an expression has different units to the symbol. In this case, the units of expression and the expected units are reported. When an inconsistency occurs within an expression, the error message will report the operation or function involved, but will not be able to identify which of several similar operations is involved.

When an old case file is read in, the only items that have units will be the in-built parameters (e.g. TransferRate) and so warning message will be created for all expressions setting these. When warnings are issued, the calculation will proceed as in earlier versions.

Once units have been added, units errors should be taken seriously (not switched off by making them warnings or ignoring them) as they could result in incorrect results.

4.3.4 Units and Availability Schemes

The issue of units in availability schemes needs careful consideration. When shared availability (i.e. elemental solubility limits) is used there is scope for errors over units.

AMBER allows the user to select the unit of amount. By default moles are used, but Bq, kg or g can be selected. The solver always works in moles, and converts the amounts (and source fluxes) between the user-units and the solver-units. This is to ensure that

the decay is handled correctly (one mole of parent decays to one mole of daughter).

Thus, all amounts that the user specifies (i.e. initial amounts and specified amounts) and source fluxes are given in user units. It is therefore natural for the units of the Limit for availability to be in the same units. For individual limits this causes no difficulty, since each item is scaled separately. For shared limits, there is a problem, because the unit conversion factors may differ between sister contaminants.

For kg or g units, the problem is not likely to be significant – sister isotopes will in general have similar atomic masses. For activity (Bq) units, the scheme will not behave sensibly – the total activity of a particular element is not of interest.

In order to minimise the likelihood of errors, while maintaining flexibility, the user must confirm the units that apply for each Limit. The default is for these to be moles (regardless of the generally applicable amount units), with the option of being in the user-defined amount units. The latter may be moles also, in which case the choice is unimportant.

Note that the expressions given for availability schemes (*Section 7.3*) must have compatible units for the choice that is made. Also, they cannot simply be constants – the unit must be explicitly stated.

4.4 Descriptions

A description can be added to any of the following:

- ▲ the case (see *Section 3.2.1*);
- ▲ compartments(see *Section 5.1.1*);

- ▲ sources (see *Section 5.1.2*);
- ▲ transfers(see *Section 5.1.3*);
- ▲ contaminants(see *Section 5.2*);
- ▲ decays(see *Section 5.3*);
- ▲ parameters(see *Section 5.4*);
- ▲ mappings(see *Section 5.4.9*);
- ▲ submodels (see *Section 7.2*);
- ▲ preconditions (see *Section 7.4*).

A description has no effect on the calculation and is purely a GUI feature to help the user.

Descriptions are free format text and can contain carriage returns. However it is not permitted to use the “(double quote) character in a description.

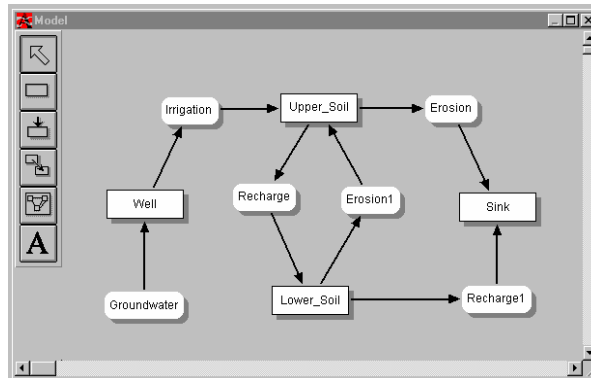
Section
5

5. Setting Up a Case File

5.1 Model Window

The Model window, as illustrated in *Figure 5.1-1*, is shown when *Windows | Model* is selected from the menu bar. It is the window in which the compartments, transfers and sources of the model are created, modified and deleted.

Figure 5.1-1
Model Window



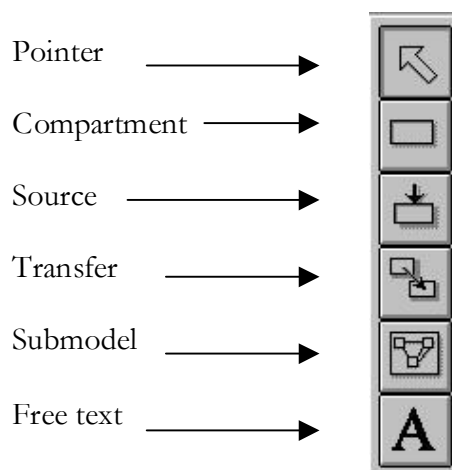
If the model being viewed is particularly large, it may be useful to use <Page Up> and <Page Down> to move up and down the view in the Model window.

A tool palette is positioned at the upper left of the window (see *Figure 5.1-2*). It contains the Pointer, Compartment, Source and Transfer tools. The tools are used to create the compartment model. The pointer tool is used to select, deselect or move any compartment, source, transfer or submodel (referred to collectively as components) in the Model window.

Adding a new compartment, source or transfer component is described in the following sections. Adding a submodel is described in *Section 7.2*.

To select an existing component, click on the pointer tool then click on the required component in the Model window. The component will become highlighted to show it has been selected. The component can then be moved by dragging it with the mouse. Changing the selection to another component can be done by clicking on the new compartment. To deselect the selected component simply click anywhere within the Model window, which does not contain a component.

Figure 5.1-2
The Tool
Palette



5.1.1 Adding a Compartment

Click on the compartment tool and then click the mouse pointer in the position on the Model window that the compartment is required. The compartment is displayed as a rectangle containing the compartment name and centred on the selected position. The compartment properties can be set by selecting the pointer tool and double clicking on the compartment box.

5.1.2 Adding a Source

Click on the source tool, click and hold down the mouse in the position for the source, drag to the receptor compartment and release. A source is displayed as a rounded rectangle containing the source name joined to the receptor compartment by an arrow. Any number of sources can be attached to a compartment. The source properties can be set by selecting the pointer tool and double clicking on the source box. **The source tool is only available if there is at least one compartment in the model.**

5.1.3 Adding a Transfer

Click on the transfer tool, click **and hold down** the mouse in the donor compartment, drag to the receptor compartment and release. The transfer is displayed as a rounded rectangle containing the transfer name joined to the two compartments by an arrow in the direction of the transfer. Any number of transfers can be attached to a compartment. Note that if multiple transfers are created between the same two compartments the transfers may be placed on top of each other. This is easily corrected by moving one of the transfer boxes. The transfer properties can be set by selecting the pointer tool and clicking on the transfer box. **The transfer tool is only available if there are at least two compartments in the model.**

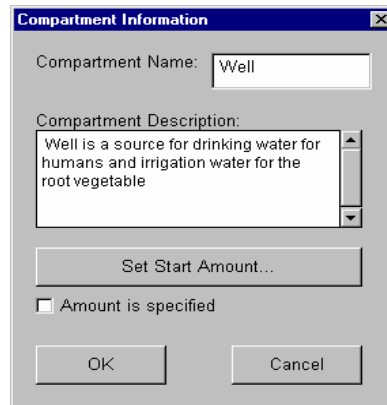
When creating new transfers and sources you must be accurate in selecting the receptor and donor compartments. If you 'miss' no new transfer will be created.

5.1.4 Compartment Properties

When the pointer tool is selected and a compartment, transfer or source is double clicked, an appropriate edit dialog box will appear (for example *Figure 5.1-3*). A new compartment, transfer or source is given a unique default name. The default name can be edited. See *Section 4.1* on 'Naming' for more information on names.

Figure 5.1-3

The
Compartment
Information
Dialog



The Compartment Information dialog (see *Figure 5.1-3*) has a text entry field for its name (which will appear in the box in the Model window) and a description field in which a description of the compartment can be given. The description can be edited by the user. It is purely informative and is not used during the calculation. The start amount can be set by clicking on the ‘Set Start Amount...’ button. The Set Start Amount dialog (see *Figure 5.1-4*) will now appear in which the start amount can be set for each contaminant (if at least one contaminant has been created). The start amount for a contaminant is set by clicking in the text entry field to the right of the contaminant. Enter the value or expression at the cursor point and press the ‘tick’ box or press the <down arrow> followed by ‘OK’.

Note that a changed cell value can be entered in the following ways:

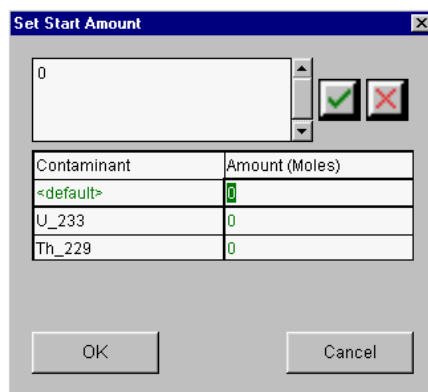
- ▲ clicking on the ‘tick’ button;
 - ▲ clicking on another cell
- and then pressing ‘OK’ in the dialog.

A changed cell value can be cancelled by:

- ▲ clicking on the 'cross' button.

If the start amount is the same for many contaminants within a compartment then the 'default' can be used. The default value for the start amount of each contaminant can be set by clicking in the text entry field to the right of 'default', entering text and clicking the 'tick' box or pressing the <down arrow>. This sets all the initial amounts for each contaminant to the default value. The default can be over-ridden by entering the required value for that contaminant as described above.

Figure 5.1-4
The Set Start Amount Dialog



When a new compartment is created the default start amount is zero. Note that the start amount cannot be time-dependent, i.e. none of its values must directly or indirectly depend on a time-dependent parameter. Once all the above properties have been set, the 'OK' button should be clicked and the compartment information will be updated. If you want to discard the changes you have made click the 'Cancel' button or use the close button.

The amount in a compartment can be specified as a function of time. This is achieved by using a check box on the Compartment edit dialog. Setting the “Amount is Specified” check box indicates that the amounts for this compartment are not to be calculated. Instead they are taken from the appropriate entries in the editable internal parameter SpecifiedAmount (indexed over Contaminants and Compartments).

When the “Amount is Specified” check box is selected the button label changes from “Set Start Amount...” to “Set Specified Amount...” and enables the amounts to be specified. Note that it is not possible to specify amounts for some Contaminants and calculate them for others in the same compartment.

When the amounts for a compartment are specified, all decays in that compartment are ignored. Transfers into and out of the compartment still affect the compartment at the other end, but have no effect on the specified compartment.

Visually, compartments with specified amounts are drawn with a shaded triangular region in the top-left corner.

5.1.5 Moving, Deleting and Editing Contaminants, Sources and Transfers

A component can be moved by clicking on the component to be moved, holding down the button and dragging the component to its new position.

A component can be deleted from the model by selecting it and choosing *Edit | Delete* from the menu bar. If a compartment box is selected and deleted, any associated sources or transfers to or from this compartment will also be deleted.

Multiple components can be selected for moving or deletion. This is done by selecting the pointer tool and clicking on the components whilst holding down <shift>. The components can now all be deleted by selecting *Edit | Delete* from the menu bar or moved by dragging them to the new position. Alternatively changes can be made to non-graphical aspects of compartments, transfers and sources, through the edit dialogs which can be accessed via windows that simply list the items. These are available from the Windows menu. Items can also be deleted from these lists, but cannot be added.

The Transfer Information dialog (see *Figure 5.1-5*) has a text entry field for its name (which will appear in the box in the Model window) and a description field in which the details of the parameter may be explained. The transfer can be set as depleting or non-depleting by clicking the radio button for the appropriate setting. A 'depleting' transfer is one where the amount of a contaminant lost from the donor compartment is equal to the amount gained by the donor compartment. For a 'non-depleting' transfer there is no loss from the donor compartment (and consequently no mass conservation).

Removal from the system can be modelled by creating a transfer to an extra 'sink' compartment.

The transfer rates (Default units are year⁻¹) can be set by clicking the 'Set Transfer Rate...' button. The transfer rates are set in a similar way to the start amount for a compartment. However, the transfer rate may be time-dependent, i.e. refer to time-dependent parameters. Once all the above properties have been set, the 'OK' button should be clicked and the transfer information will be updated. If you want to discard the changes you have made click the 'Cancel' button.

Once the transfer has been created it is still possible to change the “from” and/or “to” compartment without having to delete it and add another, by pressing the ‘new from/to’ button.

It is allowable to change the from and to compartment to one in a different submodel; but doing so may cause the transfer to “disappear” from the model window where it was selected, but all rate expressions and other information will be maintained.

The ‘set scheme’ button allows the transfer flux to be limited by the availability of a contaminant, e.g. due to solubility limitation and hence allow non-linear behaviour, see *Section 7.3, Availability Schemes*.

Figure 5.1-5
The Transfer
Information
Dialog

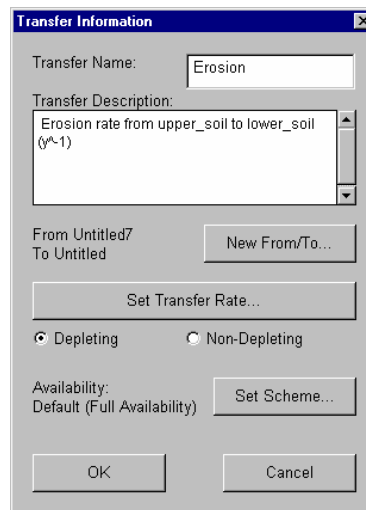
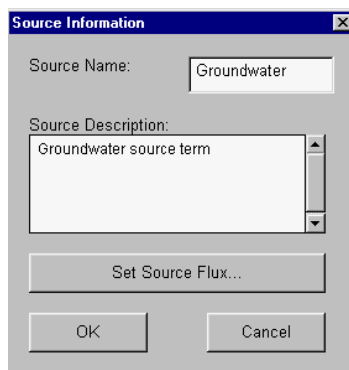


Figure 5.1-6

The Source Information Dialog

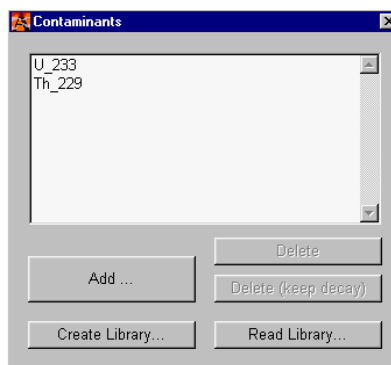


The Source Information dialog (see *Figure 5.1-6*) is similar to the Transfer Information dialog, a text entry field is given for the name and a description of the source. The source flux (for example moles year⁻¹) can be set by clicking the ‘Set Source Flux...’ button. The source fluxes are set in a similar way to the start amount. However, the source flux, may be time-dependent, i.e. refer to time-dependent parameters. Once all the above properties have been set, the ‘OK’ button should be clicked and the sources will be updated. If you want to discard the changes you have made click on the ‘Cancel’ button.

5.2 Contaminants

The Contaminants window (see *Figure 5.2-1*) is shown when *Windows | Contaminants* is selected from the menu bar or the Contaminants icon is selected from the toolbar. It is the window from which contaminants are created, modified and deleted.

Figure 5.2-1
The
Contaminants
Window

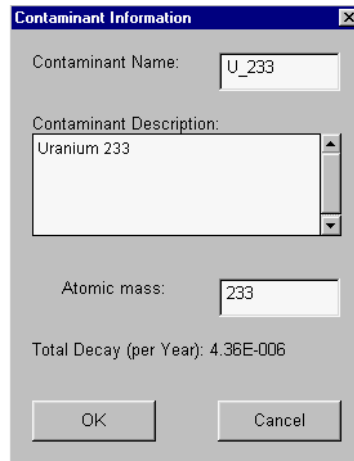


5.2.1 Adding a Contaminant

A new contaminant can be created by clicking the 'Add...' button. The New Contaminant dialog will appear (see *Figure 5.2-2*) with fields for the name and a textual description of the contaminant and its atomic mass (dimensionless).

The atomic mass is used by the 'KgToMole' and 'MoleToKg' internal parameters. The total decay rate of the contaminant is also displayed, but cannot be modified from this. Click on 'OK' to save your changes (and add the new contaminant to the Contaminants windows) and 'Cancel' to discard them.

Figure 5.2-2
The New
Contaminant
Dialog



5.2.2 Modifying a Contaminant

To modify an existing contaminant double click on its name in the Contaminants window. The Contaminant dialog will appear.

5.2.3 Deleting a Contaminant

To delete a contaminant, click on the name in the Contaminants window and then click the 'Delete' button. If *Options | Confirm deletes* is toggled, a confirmation will be required before the deletion takes place. When a contaminant is deleted all the decays that have that contaminant as a parent or a daughter will be deleted. Also all initial amounts, transfer rates and source fluxes relating to that contaminant will be deleted.

5.2.3.1 Delete (Keep Decays)

This allows individual (or multiple) contaminants to be deleted while keeping any decays for which they are daughters (by reassigning the daughter as NULL).

5.2.4 Contaminant Libraries

If a set of contaminants are used regularly, a library file can be created using the create library option. To make a library file from a set of contaminants in a case, click on the 'Create Library...' button. Another window will appear in which the user selects a filename and directory for the library file. This is done by entering the filename directly into the text field given or selecting a filename from another directory.

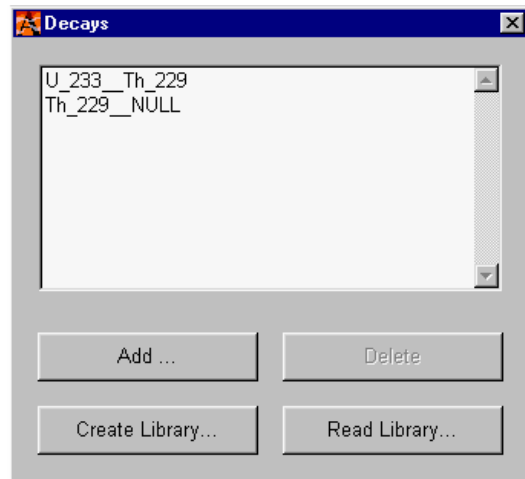
A library file or case file can also be read into the Contaminants window by clicking on the 'Read Library...' button. If a library or case file is now selected and the 'OK' button is clicked, the 'Read Library - contaminants' dialog will appear. This dialog gives information on the file being read in and the option to replace existing contaminants, merge with existing contaminants or cancel the operation. By clicking the appropriate button, the contaminants will be read into the Contaminants window.

5.3 Decays

The Decays Window (see *Figure 5.3-1*) is shown when *Windows | Decays* is selected from the menu bar or the Decays icon is chosen from the toolbar. It is the window in which decays are created, modified and deleted.

Figure 5.3-1

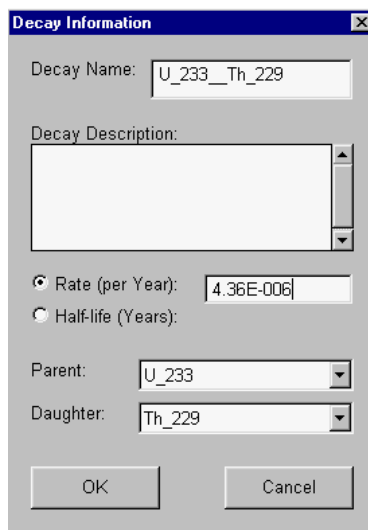
The Decays
Window



5.3.1 Adding a Decay

A new decay can be created by clicking the 'Add...' button. The New Decay dialog will appear with a field for the decay rate (for example year⁻¹) and drop down list boxes for the parent and daughter contaminants (see *Figure 5.3-2*).

Figure 5.3-2
The New
Decay Dialog



The decay rate is equal to $\ln 2 / \text{half-life}$. The decay rate must be greater than or equal to zero.

The daughter contaminant can be set to 'NULL', which indicates that the daughter contaminant is not of interest. By default the parent and daughter are specified as the same contaminant. This is clearly not possible, hence it is necessary to modify at least one of the default selections. Click on 'OK' to add the new contaminant to the Decays window.

5.3.1.1 Half-Life

The half-life of a radionuclide can be specified instead of the decay rate. This is purely a GUI feature, so AMBER immediately translates the given half-life into a rate.

5.3.2 Modifying a Decay

To modify an existing decay double click on its name in the Decays window. The Decays dialog will appear.

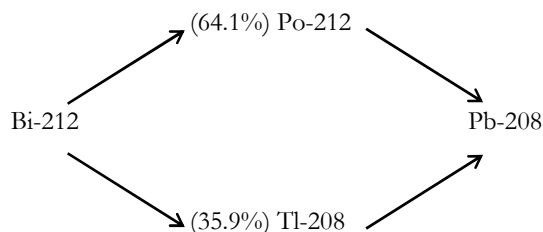
5.3.3 Deleting a Decay

To delete a decay, click on the name in the Decays window and then click the 'Delete' button. If *Option | Confirm deletes* is toggled on a confirmation will be required before the deletion takes place.

5.3.4 Decay Properties

Branching and rejoining decay chains are allowed. Decay into more than one daughter is handled by the user setting the decay rate to reflect the probability of each decay occurring i.e. the relative probability of A decaying to B or C is not given explicitly, but is implicit in the decay rates given for the $A \rightarrow B$ and $A \rightarrow C$ decay processes. It is also valid to have A decaying to B and B decaying to A. Note that it is valid to have two decays with the same parent and daughter, but that the names must be different (and so cannot both be the default name).

For example Bi-212 undergoes beta and alpha decay to give two daughter nuclides, Po-212 and Tl-208 respectively, both of which subsequently decay to give the stable isotope Pb-208.



The half-life of Bi-212 is 1.01 hours which is 1.15×10^{-4} years so the decay constant is given by $\ln 2 / (1.15 \times 10^{-4})$. However the rate of production of the two daughter nuclides will differ from the overall rate of decay of the parent nuclide and it is necessary to take into account the branching ratios for each decay (0.641 for decay to Po-212 and 0.359 for decay to Tl-208). The decay rates for production of the daughters will be given by $(0.641 * \ln 2 / 1.15 \times 10^{-4})$ and $(0.359 * \ln 2 / 1.15 \times 10^{-4})$ for Po-212 and Tl-208 respectively. The decay rates for each decay process are entered separately and AMBER calculates the overall decay rate of the parent nuclide as the sum of the branching decays. Similarly the total rate of production of the Pb-208 is calculated by AMBER following the input of the decay rates for Po-212 and Tl-208 after the user has specified that both these nuclides decay to give Pb-208.

5.3.5 Decay Libraries

If a set of decays is used regularly, a library file can be created using the create library option. To make a library file from a set of decays in a case, first click on the 'Create Library...' button. Another window will appear in which the user selects a filename and directory for the library file. This is done by entering the filename directly into the text field given or selecting a filename from another directory.

A library file or case file can be read into the Decays window by clicking on the 'Read Library...' button. If a library or case file is now selected and the 'OK' button is clicked, the 'Read Library - decays' dialog appears. This dialog gives information on the file being read in and the option to replace existing decays, merge with existing decays or cancel the operation. By clicking the appropriate button, the decays will be read into the Decays window.

5.3.6 Compartment-Dependent Decays

The decay rate can be allowed to vary from compartment to compartment. This will allow non-radioactive decay processes (e.g. organic degradation) to be modelled, and has some other useful applications.

In order to make this possible, there is a Decay NameSet. This implies that decays have names. For a case file from a version of AMBER before Version 4.1, the decay name is taken from the parent and daughter names, with a pair of underscore characters between them. This scheme is also used for new decays if no name is specified.

The decay rate set in the decay dialog is used in all compartments unless further action is taken.

DecayRate is an internal read-only parameter indexed over decays. It can be used in expressions and evaluates to the decay rate as specified in the decay dialog.

LocalDecayRate is an internal editable parameter indexed over Compartments and Decays. The default expression for LocalDecayRate is that it is equal to DecayRate in all compartments.

The solvers use LocalDecayRate to evaluate the decay rates to use. Thus, if the default is used, the same rates will be used in every compartment.

In order to change the rate in a particular compartment, the appropriate entry in the LocalDecayRate matrix must be set. This can be set to a constant or an expression as for any parameter. It can even be time-dependent and include sampled parameters if required.

This capability can be used in many ways.

For instance, by setting the decay rate to zero in the decay dialog, a decay could be switched on only in certain compartments. This could be used to model chemical processes that occur in certain circumstances only, such as organic degradation.

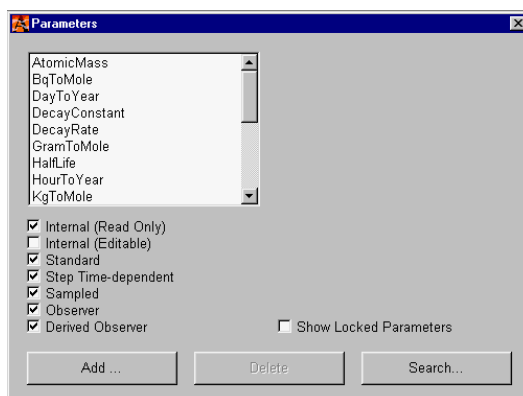
An integral of the amount in any compartment could be calculated by creating a new compartment with all decays set to zero and sending a non-depleting transfer to it with a rate set to unity.

By setting a default expression in LocalDecayRate for every decay, all compartments would use the same rate but this can be specified as an expression, which is not allowed in the decay dialog.

5.4 Parameters

The Parameters window (see *Figure 5.4-1*) is shown when *Windows | Parameters* is selected from the menu bar or the Parameters icon is selected from the toolbar. It is the window in which parameters are created, modified and deleted.

Figure 5.4-1
The Parameters
Window



Parameters are listed in the Parameters window in alphabetical order, upper case first. The check boxes on the window can be used to show which types of parameter are displayed. For example, to view only observer parameters the check boxes for all other parameter types should be deselected using a single click of the mouse. This leaves only observer parameters displayed on the screen.

The Parameter window shows a summary of the selected parameter - including multiplicity information and user description. This makes searching for particular parameters much quicker.

A standard parameter may reference standard, sampled or time-dependent parameters and may have multiplicity (see *Section 5.6* for more on multiplicity).

A step time-dependent parameter can change its value over time in a stepwise constant fashion. It does not have multiplicity.

A sampled parameter has a single value for each sample in a calculation. This value is determined by the characteristics of the sampled parameter using a random number generator. It does not have multiplicity.

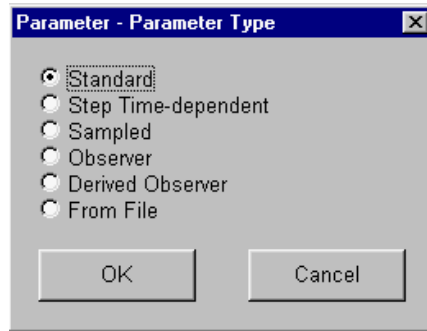
An observer parameter may reference standard, observer, sampled or time-dependent parameters and may have multiplicity.

Derived observers are a special type of observer that process the time-dependent value of another observer and calculate a result, e.g. the maximum value.

5.4.1 Adding a Parameter

A new parameter can be created by clicking the ‘Add...’ button. A dialog will appear prompting the user to select a parameter type (see *Figure 5.4-2*). Click the radio button appropriate to the type of parameter you want to create and then click the ‘OK’ button.

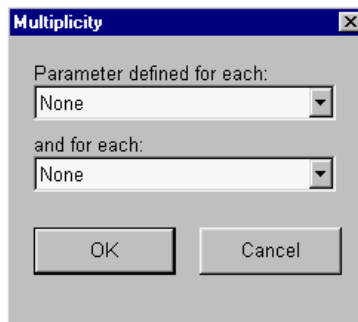
Figure 5.4-2
The Parameter Type Dialog



5.4.1.1 Standard Parameters

When a ‘standard’ parameter is created a dialog will appear (see *Figure 5.4-3*) asking for the multiplicity of the parameter (see *Section 5.6* on Multiplicity)

Figure 5.4-3
The Multiplicity Dialog



Users may specify multiplicity of parameters over internally defined NameSets - Contaminants, Compartments, Transfers and Sources - or they may construct their own NameSets as described in *Section 5.4.8*. When specifying multiplicity for parameters, the user can choose from all available NameSets, both the internal and user-defined ones. The only restriction is that a parameter indexed over two NameSets must be indexed over two different NameSets.

The allowed multiplicities are therefore:

- ▲ none (scalar);
- ▲ contaminant;
- ▲ compartment;
- ▲ transfer;
- ▲ source;
- ▲ user-defined NameSets;
- ▲ combinations of any two of these.

Select the desired multiplicity and click on the 'OK' button. The New Parameter dialog will now appear in which you can set the name, description and value(s) for the parameter. The form of this dialog will depend on the multiplicity, as illustrated in *Figures 5.4.4 to 5.4-6*.

Figure 5.4-4

The New
Parameter
Dialog: No
Multiplicity

Parameter Name: AS

Units: m2

Parameter Description: Area of upper soil (m2) over which irrigation occurs

Definition: 10000

Buttons: OK, Cancel

Figure 5.4-5

The New
Parameter
Dialog:
Contaminant
Multiplicity

Parameter Name: D_ing

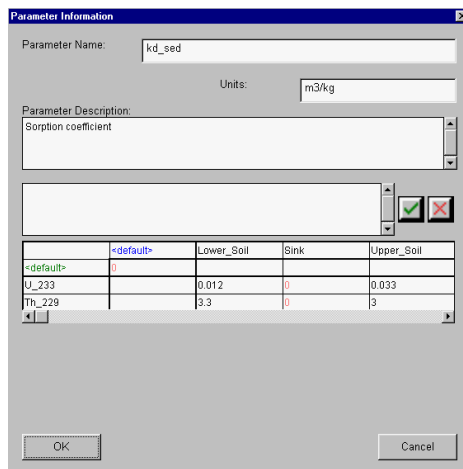
Units: SwBq

Parameter Description: Dose conversion factor for ingestion

Contaminants	Value
<default>	7.1e-008
U_233	7.1e-008
Th_229	1.1e-006

Buttons: OK, Cancel

Figure 5.4-6
 The New
 Parameter
 Dialog:
 Contaminant and
 Compartment
 Multiplicity



For parameters with a multiplicity of 1 or 2, default fields are provided. These should be used for efficiency and to save typing the same expression many times. The defaults can be overridden as desired (see *Section 5.4.2*).

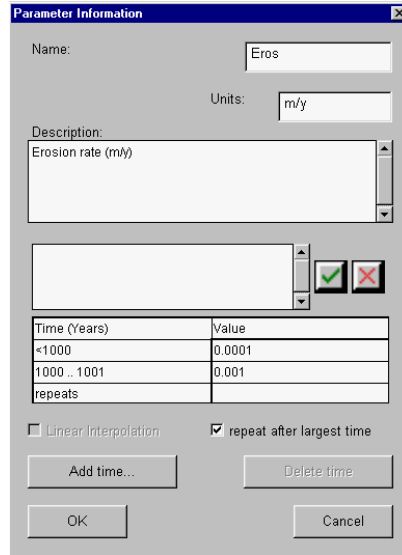
The multiplicity of a parameter cannot be changed once it has been created. However you can delete the parameter and create another one with the same name. Click on 'OK' to complete your changes or 'Cancel' to discard them.

5.4.1.2 Step Time-Dependent Parameters

AMBER permits you to specify some parameters to be time-dependent, with their values changing at specified switch times. In any specified time period (between two switch times), parameter values may be specified as fixed numeric values or algebraic expressions.

When a step time-dependent parameter is created the form of the New Parameter dialog appropriate to a time-dependent parameter will appear (see *Figure 5.4-7*).

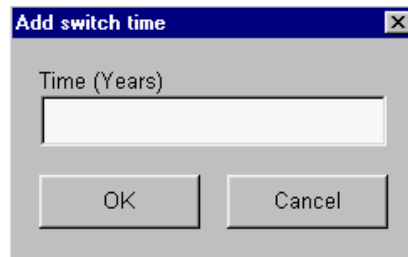
Figure 5.4-7
The New Parameter Dialog: Time-Dependent Parameter



When a time-dependent parameter is first created it has value 0 for all time. If you wish to add a 'switch time' (a time at which the parameter changes value) click on 'Add time...'.

The 'Add switch time' dialog will appear (see *Figure 5.4-8*):

Figure 5.4-8
The Add Switch Time Dialog



In the time field enter the time at which you wish the value of the parameter to change. Repeat this process until you have added all the required switch times. To set the value in the interval between switch times click on the appropriate value field and enter a constant for the value.

You can delete a switch time by clicking on the appropriate value field and then clicking on the 'Delete time' button.

5.4.1.3 Cyclic Time-Dependent Parameters

In some circumstances it is desirable to have a parameter which is piecewise-constant in time and goes through repeating cycles. Such cyclic time-dependent parameters could be used to represent dependence on day/night or on season.

To create a cyclic parameter, proceed to the New Parameter dialog for a time-dependent parameter (see *Figure 5.4-7*) and specify 'Switch Times' in the usual way. Notice that this edit dialog has check box marked 'repeat after largest time'. For cyclic parameters, this should be checked.

For example, *Figure 5.4-10* shows the Parameter Information Dialog for Eros, a parameter modelling the change in erosion rate caused by a flood every 1000 years. Eros takes two values - one for the first 1000 years, and another for year 1001. Taking the time units to be years, switch times have been added at 1000 and 1001 years and the 'repeat after largest time' box has been checked. The same values will then be used for Eros for each 1000 year cycle.

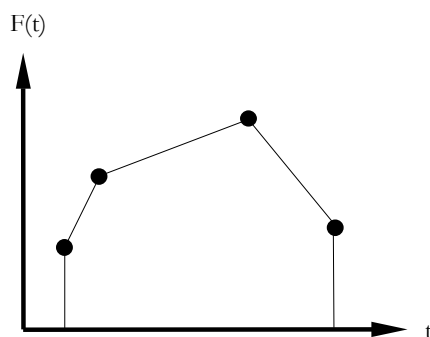
5.4.1.4 Piece-Wise Time-Dependent Parameter

The time-dependent parameters (which are specified in a piece-wise fashion) include linear interpolation. The edit dialog for time-dependent parameters has a check box for this, labelled

Linear Interpolation. This cannot be combined with the cyclic (repeat after largest time) option.

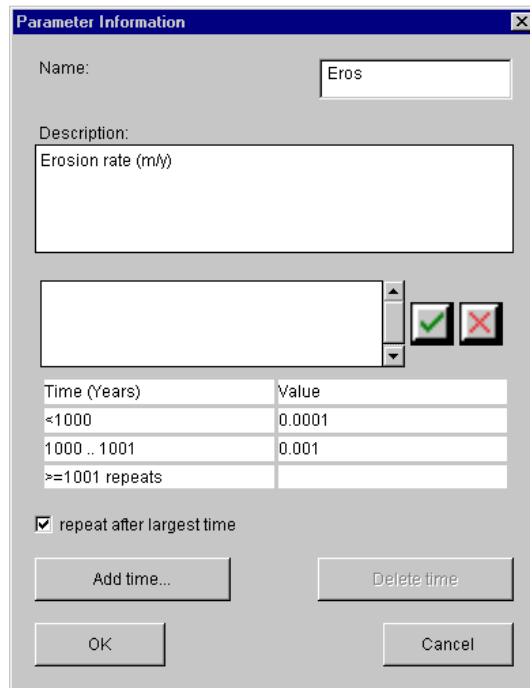
To specify the piecewise linear function, a sequence of times and expressions for each time must be given. Where the expressions are not themselves time dependent, a function like that shown in *Figure 5.4-9* will be obtained. If the expressions themselves are functions of time, then the interpolation will be applied at each intermediate point, and will therefore gradually move from one form to the other.

Figure 5.4-9
A Piecewise
Linear
Function of
Time



In combination with the Specified Amount option, this can be used to set a compartment amount as a general function of time. Note that the solvers will treat each time-point as a discontinuity and so specifying a very large number of these is not recommended.

Figure 5.4-10
The Parameter Information Dialog



5.4.1.5 Switch Parameters

Where it is known that there are discontinuities in the parameters at particular times, these should be communicated to the solver. Otherwise the solver may behave poorly, or even give incorrect results (e.g. missing a sharp spike source or transfer).

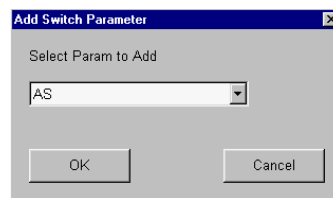
One way of introducing discontinuities is to use the time-dependent parameter type described above. A time-dependent parameter has discontinuities at given times that are used by the solver.

However, in many cases the time at which the discontinuity occurs is not constant but is a parameter. Functions such as topHat,

stepUp and stepDown can be used to implement this type of discontinuity. These functions are defined in *Table 5.5-3*. Parameters whose values correspond to discontinuities are called ‘switch parameters’ in AMBER and will be used by the solver. Note that the use of topHat, stepUp or stepDown functions of time implies the use of the time-step solver, but if the functions imply a piecewise constant set of parameters the Laplace solver can still be used, see *Section 6.2.2.1*.

To declare that a parameter is a switch parameter, select *Windows | Switch Parameters* from the menu bar or the switch Parameters icon from the toolbar. The Switch Parameters window will appear, containing a list of the switch parameters currently set. Parameters can be added to this list by clicking on the ‘Add’ button then selecting an existing parameter from the list which is then displayed in the ‘Add Switch Parameter’ dialog (see *Figure 5.4-11*).

Figure 5.4-11
The Add Switch Parameter Dialog



A parameter can be removed from the list of switch parameters by selecting it then clicking ‘Remove’. (Note that this does not remove the parameter, it simply prevents its values being used as discontinuities by the solver.)

For the topHat function, both the times for switching on and off the discontinuity should be set through parameters that are registered as switch parameters.

Any parameter with a scalar value can be used as a switch parameter. This includes sampled parameters so that random events can be modelled successfully. For instance, consider a transfer that switches on at a random time t_{Jump} . The transfer rate could be defined as

$$\text{stepUp}(t, t_{\text{Jump}}) * \text{FinalRate}$$

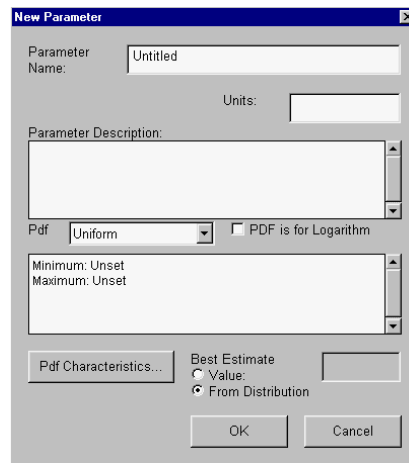
where t_{Jump} is a sampled parameter. Then t_{Jump} should be declared as a switch parameter.

Switch parameters should have units of time.

5.4.1.6 Sampled Parameters

When a sampled parameter is added, the New Parameter dialog for sampled parameters will appear (see *Figure 5.4-12*).

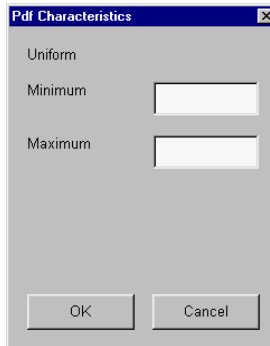
Figure 5.4-12
The New
Parameter
Dialog: Sampled
Parameter



The name and description fields can be set as with other parameters. Select a distribution type from the *Pdf* drop-down list. To set the distribution characteristics click on the 'Pdf

Characteristics...' button. This will bring up the 'Pdf characteristics' dialog. The fields in this dialog will vary depending on the type of distribution selected (for example *Figure 5.4-13*).

Figure 5.4-13
The Pdf Characteristics Dialog (Uniform Distribution)



To enable one case file for both probabilistic and deterministic models, every sampled parameter has an associated “best estimate” value.

By default, the best estimate value is calculated from the pdf characteristics. The choice of value depends on the type of pdf, see *Table 5.4-1*.

If the default is not suitable, a specific value can be given. These choices are made on the sampled parameter dialog which appears when a sampled parameter is created or edited.

Table 5.4-1
The Pdf's Available for Use in AMBER

PDF Type	Definition of default best estimate
Uniform	Mean (which is equal to the median)
Log Uniform	Median
Gaussian	Mean (which is equal to the median)
Truncated Gaussian	Mean of the untruncated distribution
Triangular	Peak
Beta	Mean
GeneralCDF	Median

In order to use the best estimate values in a calculation, this must be indicated on the calculation options dialog. The choice between Monte Carlo and Best Estimates is offered when there are sampled parameters in the case. If the Monte Carlo option is selected (the default), calculations are made for the indicated number of samples. If the Best Estimates option is selected, then a single calculation is made with each sampled parameter taking its best estimate value.

See *Section 6* on Sampling for more information on sampled parameters.

5.4.1.7 Observer Parameters

An observer parameter is similar to a standard parameter except that it can be defined in terms of other observer parameters and it can reference the internal result parameter 'Amount'.

An observer parameter is allowed to have the same range of multiplicities as a standard parameter.

Observer parameters will generally be used for post-processing, e.g. to calculate dose or risk.

It should be noted that AMBER can produce small negative results in some situations. These arise because of the numerical nature of

the time-stepping algorithm. In particular, when the amount of a contaminant in a compartment falls away to zero, there may be an "overshoot" to small negative amounts. Any observers that depend on amounts will pick up the same behaviour. Generally, these small results can just be considered to be zero. When producing charts the axis minimum can be reset to zero. It is also possible to use the "max" function to produce an observer that resets any negative values to 0, e.g. `max(Amount,0 [mol])`.

5.4.1.8 Derived Observers

Observers are calculated from the Amount, which is the primary AMBER result. Derived Observers provide the mechanism to create an Observer by using the time history of another observer.

Three types of Derived Observer can be created. These are denoted as AT, MAX and MAXTIME.

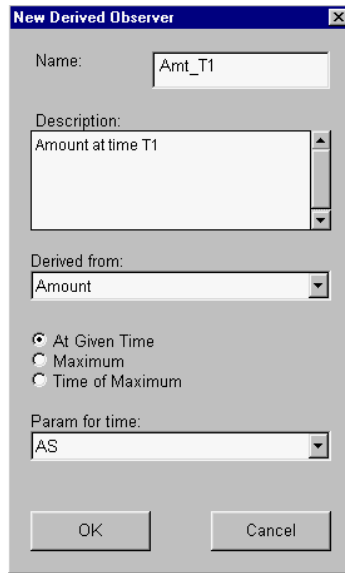
The AT version produces an Observer equal to a time-dependent parameter at a time given by another parameter (which might also be a SnapShot parameter, see *Section 5.7.5*). The MAX version gives an Observer equal to the maximum over the result times (including SnapShot times) of the parameter. MAXTIME gives the time at which this maximum occurs.

In each case the resultant Observer has the same multiplicity as the processed parameter but is not time-dependent (see *Section 5.6* on Multiplicity). The processed parameter can be an input parameter, a result (e.g. Amount) or an Observer.

The Derived Observers can be used in expressions for other Observers in the normal way. These can then be used in further Derived Observers. Thus, it is possible to set an Observer, Tmax say, to equal the MAXTIME for a particular result and then to use this to obtain a Derived Observer for all the Amounts at Tmax.

Derived Observers are added in a similar way to normal Observers. The <Add> button on the Parameters Window brings up a dialog that includes Derived Observer. Selecting this brings up a dialog in which the details of the Derived Observer can be specified. An example of a completed dialog is shown in *Figure 5.4-14*.

Figure 5.4-14
The Derived
Observer Dialog



Once created, a Derived Observer can be edited in the usual way, by double-clicking in the Parameter Window.

5.4.1.9 From File

Parameters can be created by reading a file. This has been included to allow parameter information written in another application to be read. The file can also be created manually.

By selecting the “From file” option on the “new parameter” dialog, a filename can be given (*.apf by default). This file must contain

the description for a single parameter. Once read, the user is left with the edit dialog for the parameter specified.

The contents of the file is keyword based. Valid keywords are:

NAME
 DESCRIPTION
 PDF
 LOG-PDF
 OBSERVER
 MULTIPLICITY
 VALUE

Comment lines beginning with # are ignored. Other lines are also ignored, but should be avoided.

NAME is optional. If not included then a default name will be used. If included, then the input line simply reads as follows.

NAME *parname*

DESCRIPTION is also optional. If included, the line(s) read

DESCRIPTION “a description
 which can be split across lines”

where the “...” pair delimit the description.

If PDF or LOG-PDF are included, then the parameter is a sampled variable and OBSERVER, MULTIPLICITY and VALUE cannot be specified. In this case, the full inputs are

PDF *type data* [BEST *best-estimate*]
 LOG-PDF *type data* [BEST *best-estimate*]

Where the type is Uniform, Gaussian, “Truncated Gaussian”, Triangular, or Beta.

The data that follows depends on the type.

Uniform	Min Max
Gaussian	Mean Standard-deviation
Truncated Gaussian	Mean Standard-deviation Min Max
Triangular	Min Peak Max
Beta	Min Max A B

The best-estimate is optional, and follows the keyword BEST.

The keyword OBSERVER, on a line by itself, creates an Observer rather than a Standard parameter. In either case the MULTIPLICITY keyword is used to specify the multiplicity. If omitted, a scalar parameter will be created.

The MULTIPLICITY line can indicate a one-dimensional or two-dimensional parameter.

```
MULTIPLICITY nameset
MULTIPLICITY nameset1 nameset2
```

The namesets can be internal or user defined. Internal nameset names are: Contaminants, Decays, Compartments, Transfers, and Sources.

The VALUE keyword specifies numeric values, or expressions. For parameters with multiplicity, a series of VALUE statements can be given.

Some valid VALUE lines are:

```
VALUE 1.234
VALUE "a+b"
VALUE [*] 0.0
VALUE [A][Nuc1] "x/2"
```

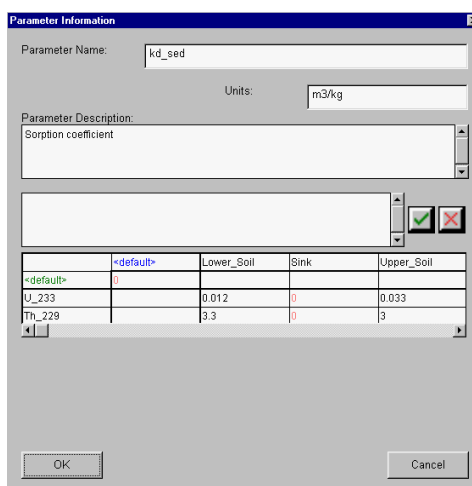
In general, the multiplicity must be followed and a number or a quoted expression.

5.4.2 Editing Parameter Values

When a parameter with multiplicity is added or edited, the user will be presented with a dialog similar to the one in *Figure 5.4-15*. The correct method for entering data into such a table is described in this section.

As the boxes in the table of parameter values are of a fixed size, a type-in box is provided where the parameter value should be entered. To add or change the value of a particular entry in the table, select the box whose value you wish to change by using the scroll bars or the arrow keys.

Figure 5.4-15
The Parameter
Information
Window



For instance, suppose you wish to edit the value of a parameter called 'retardation', with contaminant multiplicity (rows) and compartment multiplicity (columns). To change the value of retardation for one contaminant in a given compartment, select the box in this contaminant's row and the compartment's column of the table. The current value of retardation for this contaminant and compartment will appear in the type-in field where the expression or numerical value can then be edited. Click on the green tick button to incorporate the edited values into the table. When all the edits have been completed, press the 'OK' button to commit the changes. The new value of retardation for this contaminant and compartment will then replace the old one in the table.

If you decide that you do not wish to proceed with an edit at any point, press the 'cross' button and the type-in field will be cleared, with the value of retardation for this contaminant and compartment remaining unchanged in the table. If you decide you

do not want to proceed with *any* of the edits at any point, press the 'Cancel button' and all the original values will be restored.

5.4.2.1 Defaults

The entry in a parameter table labelled '<default>' provides the default value for every entry in the table (for a parameter with multiplicity greater than one, it is possible to specify a default value for a row or column.). It is more efficient to set a default value for the table and then edit the entries which differ from this.

5.4.3 Modifying a Parameter

To modify an existing parameter double click on its name in the Parameters window. The appropriate parameter dialog (depending on the type of the parameter) will appear.

When a parameter name is changed, the opportunity is automatically given to rename all the occurrences of this parameter in other expressions. This allows parameters to be renamed without having to edit many different expressions. The option to leave expressions unchanged is available. The choice is made through a simple dialog that is displayed when the name change occurs.

5.4.4 Deleting a Parameter

To delete a parameter, click on the name in the Parameters window and then click the 'Delete' button. If *Options | Confirm deletes* is toggled on, a confirmation will be required before the deletion takes place. Note that expressions that refer to a deleted parameter are not deleted. This allows, for example, a scalar parameter to be deleted and a new sampled parameter with the same name to be created without having to retype all the expressions that refer to it. If a parameter is deleted that is used in

other expressions, an additional warning message will appear, and an opportunity to cancel the deletion is given.

Multiple deletes are possible either by dragging to select a range of parameters, or holding <CTRL> whilst additional parameters are selected.

5.4.5 Internal Parameters

When a case is created a number of ‘internal’ parameters are created automatically (see *Table 5.4-2*). These internal parameters can be referenced in expressions subject to certain restrictions (e.g. referencing ‘Amount’ in an expression for the start amount of a compartment does not make sense).

In addition to these parameters are a number of internal parameters intended specifically for unit conversion (see *Section 4.3* on Units).

‘Internal’ parameters are in two groups, read-only which cannot be created, modified or deleted by the user and editable which can be modified by the user (LocalDecayRate, MinimumChangeTime, SourceFlux, SpecifiedAmount, StartAmount, and TransferRate).

Editable internal parameters can be modified via the parameter window in the same way as user-added parameters.

Table 5.4-2
Internal
Parameters

Name	Description	Multiplicity	Defined
Amount	the amount in each compartment	contaminant and compartment	after calculation
AtomicMass	the atomic mass of each contaminant	contaminant	always
Availability	the availability for each transfer	contaminant and transfer	after calculation

Table 5.4-2
Internal
Parameters

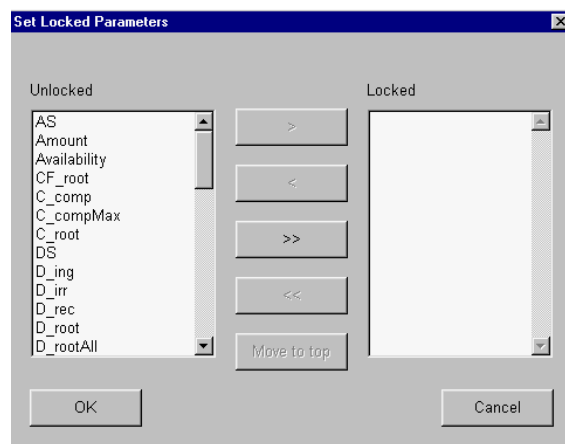
Name	Description	Multiplicity	Defined
DecayConstant	the total decay constant (rate) of each contaminant	contaminant	always
DecayRate	the rate for each decay	decay	always
HalfLife	the half life of each contaminant	contaminant	always (invalid if decay constant = 0)
LocalDecayRate	the decay rate for each compartment	compartment and decay	always
MinimumChangeTime	the time between recalculation of input parameters	none	always
SampleNumber	the sequence number of the sample (from 0)	none	always
SpecifiedAmount	the fixed amount for each contaminant in each compartment	compartment and contaminant	always
SourceFlux	the flux of all sources of each contaminant	contaminant and source	always
StartAmount	the start amount of each contaminant in each compartment	compartment and contaminant	always
TransferFlux	the flux of transfer for all contaminants and transfers	transfer and contaminant	after calculations (as it depends on the Amount)
TransferRate	the rate of transfer for all contaminants and transfers	transfer and contaminant	always
ln2	$\log_e(2)$	none	always
t	time	none	always

5.4.6 Locking Parameters

To prevent a parameter from being modified or deleted, it may be 'locked'. This facility is provided to allow many of the parameters in a complicated case to be protected against accidental deletion or modification. Individual parameters may be 'locked'.

The Set Locked Parameters dialog (see *Figure 5.4-16*) makes it simple to add or remove locks from parameters. It is accessed when *Options | Locks* is selected from the menu bar.

Figure 5.4-16
The Set Locked Parameters Dialog



To lock a parameter, select it from the list of Unlocked parameters in the Set Locked Parameters dialog then click on the single right arrow to move it into the Locked parameters list. To select several parameters from the list, hold down <control> and select the required parameters. To select a block of parameters from the list, hold down <shift> and select the required parameters. The

double right arrow will move all parameters from the unlocked list to the locked one.

To prevent locked parameters from being displayed in the full list of parameters in the Parameters window, uncheck the 'Show Locked Parameters' checkbox (see *Figure 5.4-1*).

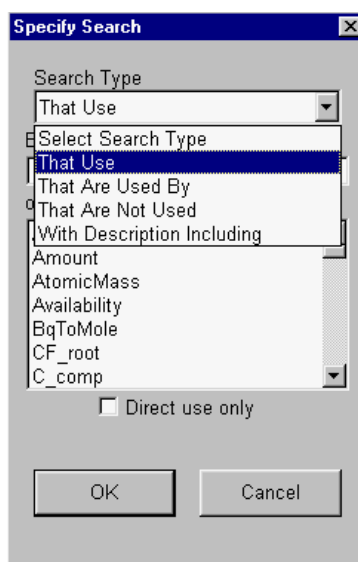
5.4.7 Parameter Searching

For large cases, keeping track of the parameters can be difficult. In order to help with this task, a parameter search capability is available.

A Search function button is present on the parameter window. This leads to a dialog that specifies the type of search to be made. The results of the search are displayed in a new parameter window.

The search dialog is shown below (*Figure 5.4-17*).

Figure 5.4-17:
The Search Dialog



The following types of search are available.

5.4.7.1 That Use

Finds all parameters that use the named parameter. The search can be restricted to direct use (that is to parameters with expressions that refer to the parameter), or finds all parameters that depend on the named parameter.

5.4.7.2 That Are Used By

Finds all the parameters that the named parameter relies on. This can be direct only (parameters appearing in expressions for the named parameter). Otherwise, finds all the parameters that are needed to define the named parameter. Note that parameters used in default expressions will be included even if the expression never actually applies.

5.4.7.3 That Are Not Used

Lists all parameters that never appear in an expression. Parameters that are not used may imply an error or redundancy, although some observers will not be used as they are intended for output purposes.

5.4.7.4 With Description Including

Finds all parameters that include the specified text string in their description text. The search can be case sensitive if required.

5.4.7.5 That Are Used in Submodel

Finds parameters that are used by a particular submodel. To count as being used by a submodel, a parameter is used by an expression for compartments, transfers or sources in the submodel.

Note that the way AMBER parameters are handled means that the list of parameters given may occasionally contain some that appear

not to be used. This occurs, for example, when default expressions are specified but not actually used. Because of this, the results of this search should be regarded as indicative only.

5.4.8 NameSets and Literals

5.4.8.1 NameSets

In AMBER, It is possible to specify multiplicity of parameters over internally defined NameSets - Contaminants, Decays, Compartments, Transfers and Sources. In addition the user can define their own NameSets.

Each NameSet has a name and a list of named members. No duplication within or between NameSets is allowed. NameSets can be created and members added to them when *Windows | NameSet* is selected from the menu bar.

To create a new NameSet, click on the 'Add...' button in the NameSet window to bring up the New NameSet dialog (see *Figure 5.4-18*).

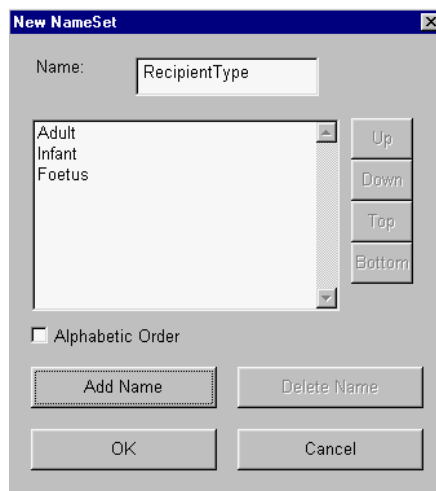
Enter the name of the NameSet. To add a member to the NameSet, click on the 'Add Name' button in the 'New NameSet' dialog (see *Figure 5.4-18*). Type the name of the member into the 'New Item Name' dialog and click on the 'OK' button. The name of the new NameSet member will appear in the list box in the 'New NameSet' dialog. Further maintenance of NameSets is performed through the NameSet window. NameSets can have as many entries as required and there are no restrictions on the number of NameSets.

Using NameSets removes the need for a set of similarly named parameters. For example, instead of three contaminant-dependent parameters called 'dose-rate-adult', 'dose-rate-infant' and 'dose-rate-foetus', a single 'dose-rate' parameter can be used. This

'dose-rate' parameter should be given multiplicity over contaminant and a NameSet (RecipientType) defined containing the members {infant, foetus, adult} (see *Figure 5.4-18*).

Figure 5.4-18

The New NameSet Dialog



When specifying multiplicity for parameters, the user can choose from all available NameSets, both the internal and user-defined ones. The only restriction is that different NameSets must be chosen for the dimensions of a parameter with multiplicity two.

Parameters with multiplicity over NameSets obey the usual rules for the use of parameters with multiplicity, For example, a parameter with multiplicity [Animals] can be divided by a parameter with multiplicity [Contaminant] to give a parameter with two dimensions - with multiplicity of both Animals and Contaminants.

5.4.8.2 Ordering in NameSets

For small cases the ordering within the NameSets is not important, since a particular item can quickly be located. For larger cases, with

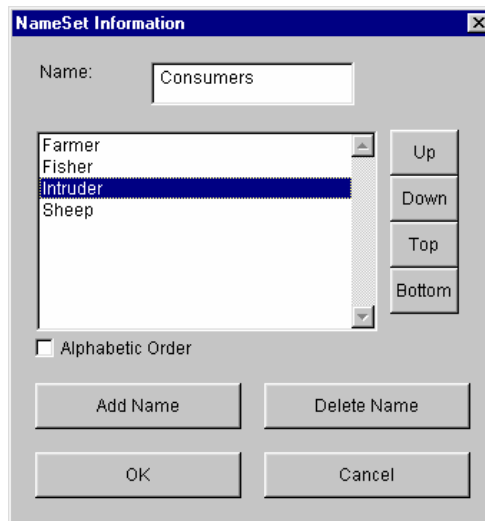
tens of compartments, this is a problem. The ordering is relevant to the user in two main ways. Firstly, in setting expressions for parameters indexed over one or more NameSets, and secondly, in selecting results to be graphed.

In earlier versions of AMBER, the ordering of items in NameSets was simply the order in which they were added. Now the user has complete control over ordering. The order can be alphabetic, creation order, or user specified. For built-in NameSets, the default is alphabetic for Compartments, Transfers and Sources, and creation order for Contaminants and Decays. These defaults can be changed by the user.

Changes to the ordering are made through the NameSet window and dialog, but are reflected in the Contaminant and Decay windows where relevant.

Ordering is controlled in the NameSet information dialog which is used for editing items in NameSets.

Figure 5.4-19:
Ordering
NameSet
Information



The Alphabetic check box should be checked if alphabetic ordering is required (See *Figure 5.4-19*). No further action is needed, as new items and renamed items will automatically be placed in the correct position.

If the alphabetic check box is not checked, then the items can be reordered as required. By default, new items are added to the end and renamed items retain their positions. The buttons allow items to be moved Up or Down or to the Top or Bottom of the list. Only one item at a time can be moved at present. To move an item, simply select it in the list and use the appropriate buttons. The item will remain selected so that a sequence of Up button clicks will have the obvious effect.

5.4.8.3 Nameset Selections

It is possible to set up a “selection” within a nameset. This can be done for user-generated or in-built namesets. The capability is accessed through the NameSets window where there is a Selections button. This allows selections to be added, deleted and edited.

Once a selection has been created, it can be used in graphing (other uses may be added in later versions). In addition to selecting “all” for the index of a graphed parameter, it is also possible to select the named selection. The relevant selections automatically appear in the dialog.

5.4.8.4 Literals

A literal is simply a word in single quotes. The word must begin with an alphabetic character and continue with alphanumeric or underscores. Thus ‘pig’ is a valid literal, as is ‘U_238’.

Literals can be used in specifying the multiplicity of a parameter, either directly or through the use of another parameter with a literal

value. Thus, if a parameter ‘Weight’ has multiplicity over the user-defined NameSet ‘Animals’, which has items {pig, cow, sheep}, then the following are valid uses:

```
Weight [‘pig’]
Weight [pig]
Weight [animal]
```

where animal is a parameter with literal value ‘pig’, ‘cow’ or ‘sheep’. Note that the second example would be ambiguous if there were a parameter pig, and this is therefore not allowed.

Literals may not be used in combination with other components of an expression. Thus, ‘pig’ + ‘U_238’ is illegal. In some cases, such errors may only be detected at run-time (i.e. the expression can be entered and saved but not used).

As a side-effect of allowing expressions in piecewise time-dependent parameters, a parameter can have a time-varying literal value. For example, it is possible to introduce a NameSet called WeatherType with members {sunny, rainy, foggy} and then to have a time-dependent parameter, ‘weather’, taking values ‘sunny’, ‘foggy’ etc. as time varies. This could then be used to set parameters whose values depend on the weather.

Note that the parameter used in an index cannot itself be indexed. Thus weight [animal [‘peter’]] is not allowed. But comparisons of literals is supported, e.g. animal == ‘cow’ is valid. If animal is the literal ‘cow’ then, the expression is true, otherwise it is false.

5.4.9 Mappings

A Mapping is a relationship between one NameSet and another. The Mapping goes “from” a NameSet “to” a second NameSet. A good example that is often used in practice is a Mapping from

Contaminants to Elements, where 'Elements' is defined as a NameSet by the user. Each item in the From-NameSet is mapped to an item in the To-NameSet. Different items in the From-NameSet can map to the same item in the To-NameSet.

AMBER generates some internal mapping that can be generally used: 'parent' is a mapping from Decays to Contaminants; 'donor' is a mapping from Transfers to Compartments, giving the "from" compartment of a transfer.

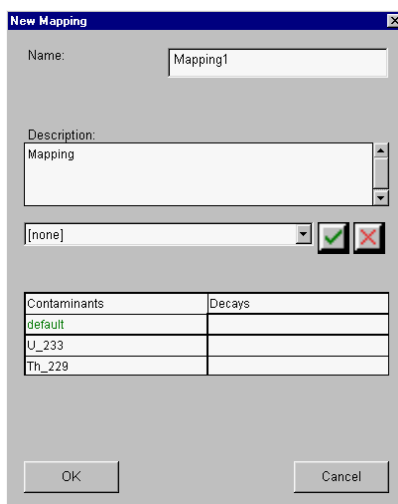
To create a mapping, select *Windows | Mappings* from the menu bar or the Mappings icon from the toolbar. The Mapping window will appear. Click on the 'Add..' button to move to the 'Create Mapping' dialog. This dialog displays two list boxes from which the From-NameSet and To-NameSet can be selected. Click on the 'OK' button to confirm your selection, and you are moved to the 'New Mapping' dialog (see *Figure 5.4-20*).

The user can then set the mapping name, give it a description and prescribe the mapping between the members of the From-NameSet and the To-NameSet.

To prescribe the mapping, select the entry in the table mapping the first member of the From-NameSet to the first element of the To-NameSet. Use the list box to select the corresponding element of the To-NameSet and click on the tick button. Click on the 'OK' button to return to the Mapping window when the complete mapping has been prescribed.

Figure 5.4-20

The New
Mapping Dialog



The following example describes how a mapping from Contaminants to Elements might be used in an expression to convert a parameter with multiplicity of the To-NameSet to give a result with multiplicity of the From-NameSet. The NameSet 'Elements' has elements {U, Th} and the mapping named 'ContaminantToElement', maps:

U-238 -> U
 U-233 -> U
 Th-229 -> Th

Suppose the sorption coefficient, Kd, has multiplicity 'Elements' and that $Kd [U] = 1$, $Kd [Th] = 0.1$. Note that the expression $Kd(\text{ContaminantToElement})$ gives a vector with values for U-238, U-234 and Th-230.

The 'Retardation', with multiplicity 'Contaminants', could then be defined in terms of the sorption coefficient as:

Retardation [*] = 1.0 + (1-PHI)/PHI * RHO* Kd
(ContaminantToElement).

It may often be more efficient (especially for time-dependent cases) to define a parameter with multiplicity over the From-NameSet that is just assigned using the mapping. This saves the mapping being applied many times. In the above example we could add a parameter ContamKd with multiplicity Contaminants, set ContamKd [*] = Kd (ContaminantToElement) and then use Retardation [*] = 1 + (1-PH1)/PH1 *RHO * ContamKd. This would be particularly beneficial if RHO was time-varying since it would avoid recalculating the mapping (assuming that Kd is time-independent).

5.5 Expressions

5.5.1 General

The use of symbolic expression provides powerful modelling functionality. Expressions can be used to define the values of:

- ▲ standard parameters;
- ▲ observer parameters;
- ▲ transfer rates;
- ▲ source fluxes;
- ▲ local decay rates;
- ▲ start amounts.

However only numeric constants may be used to define the values of:

- ▲ time-dependent parameters;
- ▲ sampled parameter characteristics;
- ▲ result times;
- ▲ global decay rates;
- ▲ atomic masses.

Spaces are not significant in expression, except where they are introduced into names, which will cause an error. For example:

AAA+BBB

is equivalent to:

AAA + BBB

However an error will be caused by the space in:

AAA + B BB

The operators and functions available are given in *Tables 5.5-1 to 5.5-4*, and the order of precedence for the operators in *Table 5.5-5*.

Table 5.5-1
Unary
Operators

Symbol	Syntax	Description
-	-x	unary minus
+	+x	unary plus

Table 5.5-2
Binary
Operators

Symbol	Syntax	Description
-	x1-x2	subtraction
+	x1+x2	addition
*	x1*x2	multiplication
/	x1/x2	division
^	x1^x2	exponentiation

Table 5.5-3
Mathematical
Functions
Available

Name	Syntax	Description
Abs	abs(x)	absolute value of x
Cos	cos(x)	cosine of x (x in radians)
Exp	exp(x)	e to the power x
Int	int(x)	integer part of x
Sqrt	sqrt(x)	square root of x
Ln	ln(x)	logarithm (base e) of x
log10	log10(x)	logarithm (base 10) of x
Min	min(x1,x2,x3,...,xN)	minimum of arguments
Max	max(x1,x2,x3,...,xN)	maximum of arguments
Sin	sin(x)	sine of x (x in radians)
StepUp	stepUp(x1,x2)	0 if $x_1 < x_2$, 1 otherwise
StepDown	stepDown(x1,x2)	0 if $x_1 > x_2$, 1 otherwise
TopHat	topHat(x1,x2,x3)	1 if $x_2 \leq x_1 \leq x_3$, 0 otherwise
Sum	sum(expression)	sum of components
Largest	largest(expression)	largest of components
LookUp Functions	lookUp(x,y0,x1,y1...)	piecewise constant
	cyclicLookUp(x,y0,x1,y1...)	periodic piecewise constant
	linearLookUp(x,x1,y1...)	linear interpolation

Table 5.5-4

Boolean
Functions
Available

Boolean Operator	Description
a>b	1 if a is greater than b and 0 otherwise.
a>=b	1 if a is greater than or equal to b and 0 otherwise.
a<b	1 if a is less than b and 0 otherwise.
a<=b	1 if a is less than or equal to b and 0 otherwise.
a==b	1 if a is equal to b and 0 otherwise.
a!=b	1 if a is not equal to b and 0 otherwise.
not(a)	1 if a is 0 (false), and 0 otherwise.
and(a,b,..)	1 if all its arguments are non-zero (true), and 0 otherwise.
or(a,b,..)	1 if any of its arguments are non-zero (true), and 0 otherwise.
xor(a,b,..)	1 if exactly one of its arguments is non-zero (true), and 0 otherwise.

Table 5.5-5

Order of
Precedence

Operator	Description	Precedence
[...]	Multiplicity	Highest
(...)	Parentheses	
- and +	unary minus and plus	
^	Exponentiation	
* and /	multiplication & division	
- and +	addition & subtraction	Lowest

5.5.2 Boolean Expressions

The Boolean operators and functions return 1 for true and 0 for false (any non-zero value is treated as true in the functions). The operators are: >, >=, <, <=, == and !=. The functions are not(), and(), or() and xor().

5.5.3 Sum and Largest Function

The 'sum' and 'largest' functions find the sums and maxima across the components of a parameter, respectively. (Note that the function names are 'sum' and 'largest' all in lower case.) Each function has two versions: one to act over the whole of a parameter and the other to act over a single dimension.

The simple form is `sum (expression)` and `largest (expression)`. It is recommended that the expression should be a simple one, often a single parameter name. In this form the action is over the full multiplicity. Thus `sum(Amount)` would result in a single value equal to the total amount over all contaminants and compartments.

The extended form specifies the index over which the sum should be performed. The result has one value for each item in the other index. Thus `sum(Amount, Compartments)` gives a vector of results, one for each Contaminant.

Note that there is no 'smallest' function as this can be simulated using `-largest(-X)`.

5.5.4 SumType Function

The `sumType` function sums over items mapping to the same target. It can be used, for example, for summing Amounts:

```
ElementAmount = sumType(Amount, element),
```

where `element` is a mapping, and the resulting `ElementAmount` would have multiplicity over Elements and Compartments.

It is useful in many other situations, e.g.

```
MaterialAmount = sumType(Amount, material),
```

where `material` is a mapping from Compartments to a Materials, or

```
ChainDose = sumType(Dose, chain),
```

where `chain` is a mapping from Contaminants to a DecayChains.

The first item can be an expression as with other AMBER functions. The result has the multiplicity of this expression except that the component corresponding to the mapping source is replaced by that corresponding to the mapping target. So, Amount is indexed over Compartments and Contaminants, if material is a mapping from Compartments to Materials, then MaterialAmount must be indexed over Materials and Contaminants.

5.5.5 TypeLookup

This is needed for the shared availability schemes, but can be used more widely.

`typeLookup(expression, mapping)`

uses the result of expression, which must be indexed over the target of mapping, and produces a result indexed over the source of mapping. Thus,

`typeLookup(Kd, element)`

gives a Kd value for each contaminant, given the Kd for each element. Note that this simple form (where the expression is a single symbol) is equivalent to the construct `Kd(element)`. The function form extends this to handle expressions.

5.5.6 LookUp Functions

Three “Look Up” functions are provided in AMBER. These are `lookUp`, `cyclicLookUp` and `linearLookUp`.

Each provides a function of their first argument (denoted *x* here) in terms of the values at a discrete set of points. The behaviour between the points differs from function to function. In all cases, the units of the *x*'s should be compatible and the units of the *y*'s should be compatible.

The `lookUp` function provides a piecewise constant value. It takes the form `lookUp(x, y0, x1, y1, x2, y2, . . . , xN, yN)` (where $x_1 < x_2 < x_3 < \dots < x_N$). The returned value is

y_0	$x < x_1$
y_1	$x_1 = x < x_2$
y_2	$x_2 = x < x_3$
...	...
y_N	$x_N = x$

The `cyclicLookUp` function provides a piecewise constant value that is periodic. It takes the form `cyclicLookUp(x, y0, x1, y1, x2, y2, . . . , xN)` (where $x_1 < x_2 < x_3 < \dots < x_N$). The returned value is

y_0	$x < x_1$
y_1	$x_1 = x < x_2$
y_2	$x_2 = x < x_3$
...	...
y_{N-1}	$x_{N-1} = x < x_N$
y_0	$x_N = x < x_N + x_1$
y_1	$x_N + x_1 = x < x_N + x_2$
...	...

The `linearLookUp` function provides a linearly interpolated value. It takes the form `linearLookUp(x, x1, y1, x2, y2, . . . , xN, yN)` (where $x_1 < x_2 < x_3 < \dots < x_N$). The returned value is

0	$x < x_1$ Or $x > x_N$
y_i	$x = x_i$
$y_i + (y_{i+1}-y_i)*(x-x_i)/(x_{i+1}-x_i)$	$x_i < x < x_{i+1}$
...	...

5.6 Multiplicity

Multiplicity is similar in concept to arrays in programming languages. A parameter with multiplicity potentially has more than one value. For example a parameter with contaminant multiplicity has a value for each contaminant.

It is possible to have multiplicity across one or two dimensions. This can either be the internal NameSets, Contaminants and Compartments, or user defined NameSets.

You can use the square bracket notation to access elements of a parameter with multiplicity. Consider a standard parameter 'concentration' with multiplicity contaminant and compartment. A standard scalar (no multiplicity) parameter 'sum1' can be created to sum concentration in compartments 'A' and 'B' for contaminant 'NUC1'. 'sum1' will have expression:

```
concentration[contaminant=NUC1, compartment=A]+
concentration[contaminant=NUC1, compartment=B]
```

This can be shortened to the equivalent:

```
concentration[NUC1,A] + concentration[NUC1,B]
```

If you want to sum the concentrations of A and B for each contaminant a standard parameter 'sum2' could be created with multiplicity contaminant and default expression:

```
concentration[A] + concentration[B]
```

Using a default expression saves having to type this same expression for each contaminant and is more efficient in calculation. Here the contaminant index is implicit, i.e. for contaminant 'NUC1' the expression will be interpreted as:

```
concentration[NUC1,A] + concentration[NUC1,B]
```

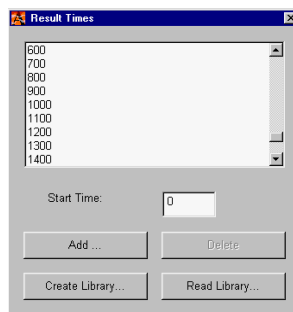
In dual multiplicity, column defaults take precedence over row defaults.

Note that the square bracket syntax cannot be confused with the use of square brackets for units since the latter is only valid after a constant. Thus x[m] means that x has an index that includes m, whereas 2[m] means 2 metres!

5.7 Result Times

The Result Times window (see *Figure 5.7-1*) is shown when *Windows | Result Times* is selected from the menu bar or the Results Times icon is selected from the menu bar. It is the window in which result times are created, modified and deleted.

Figure 5.7-1
The Result
Times window



The start time, which must be greater than zero and less than the earliest result time, is entered in the start time field provided. It is zero by default.

5.7.1 Result Time Units

The units used for result times can be different from those used for data input and rates. By default, the same units are used – consistent with earlier versions.

If different units are used for result times, then the only effect is on times specified in the Result Times Window, and on the times given in reports and graphs. All other times remain in the internal units – including the time variable itself, t .

Changing the time units (or other units) will invalidate any calculated results.

The Units Dialog is accessed from the *Options* | *Units* menu.

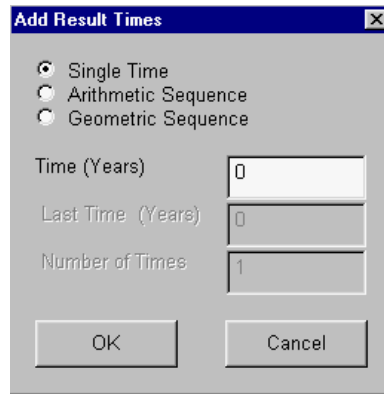
5.7.2 Adding a Result Time

One or more new result times can be created by clicking the ‘Add...’ button. The Add Result Times dialog (see *Figure 5.7-2*) will

appear with a choice of a single time or an arithmetic or geometric sequence of times and fields that you can use to specify the new times.

To add a single result time select the 'Single Time' radio button and enter the time in the 'First Time' field. To enter an arithmetic or geometric sequence of times select the 'Arithmetic Sequence' or 'Geometric Sequence' radio button, enter the first and last times in the sequence in the 'First time' and 'Last Time' fields and the number of times in the 'Number of Times' field. For example to add the sequence of result times 10,20,30...200 select 'Arithmetic Sequence', enter 10 in the 'First Time' field, 200 in the 'Last Time' field and 20 into the 'Number of Times' field. Result times are listed in increasing order. If a duplicate time is input a warning dialog appears but the duplicates are then discarded. Click on 'OK' to add the new result times to the Result Times window.

Figure 5.7-2
The Add Result
Time Dialog



5.7.3 Modifying a Result Time

To modify an existing result time double click on the result time in the Result Times window. The Change Result Time dialog will appear. Change the value in the field as required.

5.7.4 Deleting a Result Time

To delete a result time, click on it in the Result Times window and then click the 'Delete' button. If *Options | Confirm deletes* is toggled on, a confirmation will be required before the deletion takes place.

5.7.5 Reporting Results at a Sampled Time

It is sometimes necessary to be able to report results at a sampled time, which can be done by using Snapshots and Derived Observers, either in conjunction or independently of each other.

5.7.5.1 SnapShots

In general, AMBER stores the Amount in each compartment for each contaminant at each output time. Most results are then presented by connecting the results at the output times by straight lines. This is generally adequate, but cannot cope with a situation where the time of interest varies, or is derived from a calculation. This is particularly true where there is a discontinuity of some sort at or near the time of interest. In these circumstances, it is useful to be able to record the Amounts at the time of interest. This is what SnapShots are for.

A SnapShot is specified by giving a variable that evaluates to the time of interest. In this way, a SnapShot time is very similar to a Switch time. The GUI for SnapShots is identical to that for Switch Times, and is accessed via the *Windows* menu.

It should be noted that, by itself, the introduction of SnapShots does not change any results reported by the GUI. SnapShots are most useful when used in conjunction with the Derived Observer.

SnapShot parameters cannot be Observers, but can be Sampled. They must evaluate to a single value, with time units.

5.7.6 Result Time Libraries

If a set of result times are used regularly, a library file can be created using the create library option. To make a library file from a set of result times in a case, first click on the 'Create Library...' button. Another window will appear in which the user selects a filename and directory for the library file. This is done by entering the filename directly into the text field or selecting a filename from another directory.

A library file or case file can be read into the Result Times window by clicking on the 'Read Library...' button. If a library or case file is now selected and the 'OK' button is clicked, the 'Read Library - result times' dialog appears. This dialog gives information on the file being read in and the option to replace existing result times, merge with existing result times or cancel the operation. By clicking the appropriate button, the result times will be read into the Result Times window.

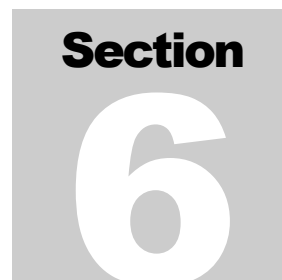
5.8 Saving the Case

It is advisable to save the case regularly, particularly after significant data has been added and prior to and following calculation. Select *File | Save Case As* from the menu bar to save the case with a new path and *File | Save Case* from the menu bar or the Save icon from the toolbar to save the case with the same path.

If a case has been altered and the *File | Save Case* option is selected, AMBER makes a backup copy of the previously saved version of the case. This is given the same name as the case file with the extension '.cbk'. This is useful if the saved case is accidentally overwritten.

If a case has been altered and the original case is required the *File | Revert to Saved* option can be used. This will delete all the changes made since the case was last saved.

If an attempt is made to close a case with unsaved changes, then a message is produced prompting for a save or discard.

A grey square containing the word "Section" in a bold, black, sans-serif font at the top, and a large, white, sans-serif number "6" in the center.

6. Sampling

In addition to calculating results for a particular set of parameter values (a deterministic run), AMBER provides facilities for performing multiple calculations by sampling the values of some parameters from probability distributions (a probabilistic run) (see *Section 5.4.1.6* on sampled parameters). There are two main reasons for undertaking a probabilistic run: uncertainty analysis; and sensitivity analysis.

In an uncertainty analysis the objective is to discover the implications of uncertainty in the values of input parameters on the calculated results. Thus, it is necessary to characterise the input parameter uncertainty, in terms of probability distributions, and calculate the distribution of results. The results can be analysed, for example to give averages (see *Sections 7.3.1* and *7.3.2* on generating reports and graphs).

In a sensitivity analysis the objective is to discover the relationship between the input parameters and particular results, in order to see which parameter values are most important. Here, the distributions given for the input parameters serve only to indicate the range of values that are of interest. The distribution of results is analysed to indicate correlation with the inputs. This functionality is provided in standard statistical packages and AMBER can produce files suitable for input to

one of these, SPSS. The details of this are discussed in *Section 7.3.3*.

It is, of course, possible to set up a probabilistic run and perform both uncertainty and sensitivity analysis on the same samples. The remainder of this section describes the various input distributions that are provided and how these can be used to produce a wide variety of parameter distributions and correlations.

The mechanisms within AMBER for user introduced parameters is ideal for undertaking probabilistic studies. This is because the ultimate sources of uncertainty can be identified and sampled parameters introduced to reflect them. This naturally leads to sensible correlation structures.

Consider a case where two parameters are uncertain because they both depend on a common third item. This is simply handled by making this common source of uncertainty a sampled parameter and giving the formulae that can be used to calculate the original two parameters. These parameters are then correlated, through their mutual dependency.

The first step in performing an uncertainty analysis is to clearly locate the sources of uncertainty (not just the symptoms of it - the derived parameters). Each of the sampled parameters must then be given a distribution. Note that in AMBER all sampled parameters are scalar. This is because attaching a distribution to a parameter indexed by contaminant (say) would be ambiguous. It could mean that all the contaminant values should be sampled from the same distribution (independently of one another) or it could be that only one value should be sampled, and used for every contaminant. In AMBER, the user is forced to resolve the ambiguity, by introducing only one or many sampled parameters, as appropriate.

Once the sampled parameters are identified it is necessary to attach a distribution to them. The art of eliciting such a distribution from experts is beyond the scope of this document, it is assumed that a particular distribution has been decided upon.

AMBER provides six basic distribution types that can be used directly. However, by using expressions involving the basic types, other distributions can be synthesised. The basic types are described below. All distributions can also be specified directly as logarithmically based distributions. This is indicated by a check box on the distribution specification dialog.

For a log-based pdf, the distribution parameters relate to the log, base 10, of the final variable. Thus, for a log-uniform from 10^{-2} to 10^{+2} , the log check box is selected and the min and max given as -2 and $+2$. Similarly, for a log-Gaussian (log-normal) distribution, the mean and standard deviation apply to the logarithm.

Note that the LOG-UNIFORM (present in AMBER versions before 4.3) distribution is now redundant. Case files using it are converted automatically to use an equivalent log-based uniform distribution.

6.1 Uniform

This requires simply a minimum and maximum value. Any intermediate value is equally likely to be sampled.

6.2 Gaussian

Gaussian, or normal, distributions are well-known, and are simply characterised by a mean (central value) and standard deviation (width). See any statistical text book for further discussion. Note that in there is no truncation (use the Truncated Gaussian if this is required), and care should be taken with using Gaussian distributions for parameters which have physical limits (e.g. they must be positive), since in principle the tail of the distribution continues indefinitely.

6.3 Truncated Gaussian

This is precisely the same as a Gaussian, except the samples outside set limits will be rejected. It is useful when a parameter distribution has a basically Gaussian shape, but where other considerations indicate that it lies within definite bounds. Note that very tight bounds are not allowed, since the time take to generate a non-rejected sample could become prohibitive.

6.4 Triangular

This simple distribution requires a minimum, a maximum and a most-probable (peak) value. Note that the probability of sampling a value lower than the peak value depends on the relative position of this between the minimum and the maximum. Care should be taken in using a triangular distribution when the range is very wide. For example, if the minimum is 1, the peak is 10 and the maximum is 100, it is not correct to suppose that half the sample will be less than 10 (in fact only 1 sample in 11 will be).

6.5 Beta

The beta distribution can be used to give a wide range of distributions, including bi-modal distributions. All values will be between the 'minimum' and the 'maximum'. The shape is determined by the 'beta A' and 'beta B' parameters. 'beta A' and 'beta B' must be >0 and, for practical purposes, <10 . If the pdf is plotted the minimum end is shaped like:

$$(x-\text{minimum})^{\text{beta A}-1}$$

and the maximum end like:

$$(\text{maximum}-x)^{\text{beta B}-1}$$

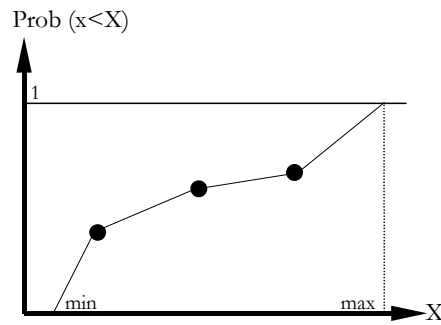
If 'beta A' and 'beta B' are the same the distribution will be symmetric.

6.6 General Probability Distribution Functions

The General CDF allows any shape of cumulative probability distribution to be approximated as piecewise linear. This corresponds to a histogram for the density function, but is slightly more general (the values are not necessarily spaced equally).

To specify the piecewise linear function, the minimum and maximum values must be specified and cumulative probabilities given at the intermediate points. This is achieved through the pdf editing dialog.

Figure 6.6-1
A Piecewise
Linear
Cumulative
Distribution
Function



6.7 Partial Sampling

Sampling some parameters while using the best-estimate value for others. This allows a deterministic run using the best-estimate to be made in a probabilistic case.

Partial sampling enables the user to select the parameters to be sampled before calculation is run. The non-sampled parameters take best-estimate values.

Note that the values taken by sampled parameters are independent from one parameter to another, so the sequence of values for any sampled parameter will be the same regardless of which other parameters are sampled. This allows comparisons to be made between results with or without varying one parameter with all other parameter values the same.

The parameters to be sampled can be specified from the Calculation Dialog. If Partial Sampling is selected, the Specify... button gives access to the list of sampled parameters. This is just a selection list, similar to those used elsewhere in AMBER. The selected items will be remembered from calculation to calculation and are saved in the case file.

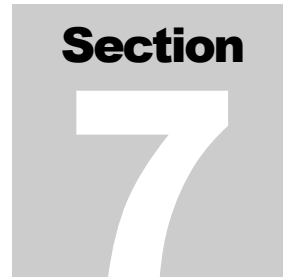
Full sampling or best-estimate calculation can be selected separately – this will not destroy the selections made. Note that if “Partial” sampling is undertaken with no sampled parameters, a set of identical best-estimate calculations will be made!

6.8 Seeds

The samples are generated in AMBER by a pseudo-random number generator. This requires an initial seed. If the initial seed is changed, a different set of samples will be generated. Note that the precise samples generated also depends on the order in which the distributions were created. When new sampled parameters are added, the sample values for the existing ones do not change. Similarly if a sampled parameter is deleted, the remaining ones will generate the same sampled values.

6.9 Selecting Specific Samples

By specifying the ‘first’ sample in the Calculation Dialog (*Section 7.2.3*) it is possible to run only particular samples. This can be used, for example, to re-run a sample that gave high consequences, to look at its detailed time-dependent behaviour in isolation. The internal SampleNumber parameter is useful in determining which particular samples are important. For example a scatter plot of sample number versus the peak consequence could be used.

A gray square graphic containing the word "Section" in a bold, black, sans-serif font at the top, and a large, white, stylized number "7" in the center.

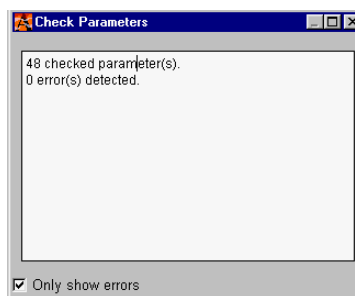
7. Running and Reporting a Case File

7.1 Checking Parameters

Calculations and report generation can sometimes result in errors being reported in a case due to incorrect parameter definitions. For example you might have defined a parameter 'x' in terms of parameter 'y', where 'y' is undefined. The Check Parameters window provides a useful way to 'debug' such problems. It is good practice to check parameters both before and after calculation to avoid any errors propagating through the results.

Select *Results* | *Check Parameters* from the menu bar. The Check Parameters window (see *Figure 7.1-1*) will appear with a summary of all the user defined parameters and model inputs. Simple evaluation errors will be reported. Observers will not be checked unless a calculation has been run.

Figure 7.1-1
The Check
Parameters
Dialog



Note that the information in the window is a ‘snapshot’. If you want to update it you need to select *Results | Check Parameters* again and AMBER will re-evaluate all the parameters each time.

If the window already exists when you select *Results | Check Parameters* it will be brought to the front and the contents updated.

The Check Parameters window can be resized and can be used for copying from. This facility can be useful if expressions for one parameter are to be copied into another. The copy facility is also useful for pasting into a word processing package or text editor for printing selected information or inserting into a report.

7.1.1 Graphing

In AMBER 4.5, input parameters can be graphed prior to a calculation. This is particularly useful for checking time-dependent and sampled inputs. Graphing is discussed in *Section 7.3*

7.2 Calculation

7.2.1 Solvers

AMBER has an internal Laplace solver and a time-step (‘Enviros Software Solutions Dylan’) solver.

Changes first made in AMBER 4.4 allow the solver to run more quickly. For example, changes have been made to reduce unnecessary copying of data, allocation of memory etc. These have no effect on the results (beyond possible trivial rounding differences when operations are applied in a different order). However, these changes have a significant impact on the run times for most AMBER calculations (particularly those using the time-stepping solver).

7.2.2 Solver Controls

Two capabilities can help in selecting and controlling which solver is to be used.

7.2.2.1 Forcing the Laplace Solver to be Used.

For cases where the transfer rates, sources, specified amounts and local decay rates are piecewise constant, but use stepUp, stepDown or topHat functions, AMBER does not recognise that the Laplace Solver could be used. The user can force the use of the Laplace solver in this situation, by selecting it on the Calculate Options dialog that comes up when a calculation is requested.

AMBER checks that the problem is suitable for Laplace solution before proceeding. The discontinuity times must be properly specified through the switches window for this option to function correctly.

7.2.2.2 Speeding up the Time-Step Solver When Rates Change Only Slowly.

When the time-step solver is being used, AMBER has to recalculate the transfer rates, sources, specified amounts and local decay rates at every time-step. For complex problems this can be a slow process. In many cases, the rates change only slowly with time and it is not necessary for them to be recalculated every time-step.

The user can speed up the solver in these cases by specifying a minimum time interval. For intervals shorter than this all the rates can be calculated by linear interpolation.

The interpolation scheme is used for transfer rates (and sources, availability parameters, decay rates and specified amount if appropriate). This is invoked by using the `MinChangeTime` parameter.

Note that the interpolation is only used between discontinuities (as set by `Switch Time` parameters). Using the interpolation scheme can substantially reduce run times without altering the results significantly.

When the Interpolation scheme is used, AMBER will try to use a linear interpolation scheme instead of calculating the full transfer rate expressions (and sources and availability parameters) at every time step. When the time-step is smaller than the `MinChangeTime`, this means that the rate will be calculated every `MinChangeTime` interval. So if `MinChangeTime` is 10 years, the rates will be calculated at 0, 10, 20 etc. Once the step is longer than 10 years then the rate will generally be calculated every step as before. `MinChangeTime` can itself be time-dependent.

It is important to note that the `MinChangeTime` should be set to the smallest time over which significant changes to

transfer rates occur, regardless of the rates themselves. Thus, if transfer rates, sources and availability parameters do not change the `MinChangeTime` can be set very large.

AMBER does not check that the given `MinimumChangeTime` is sensible – so it should be used with care. This does mean that a quick and approximate calculation can be performed by setting the `MinimumChangeTime` to a large value and ignoring the shorter timescale changes.

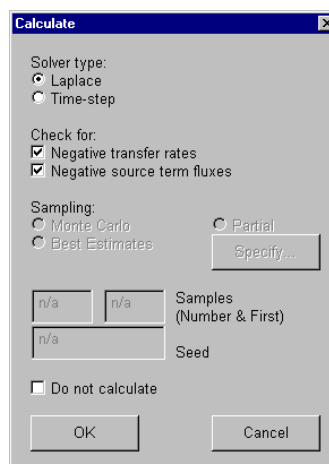
Note that earlier versions of AMBER used a piecewise constant interpolation scheme if `MinChangeTime` was set. This is no longer available since the new scheme is much more efficient.

7.2.3 Performing a Calculation

Once the model has been set up to run a calculation, the *Results | Calculate* item can be chosen from the menu bar or the Calculate icon is selected from the toolbar. The Calculate dialog will now appear (see *Figure 7.2-1*).

Either the Laplace or time-step solver will be selected. If you have used the variable 't' in expressions that are used directly or indirectly to calculate source terms and/or transfers you will be forced to use the time-step solver. If you have not used 't' (except for observer parameters) you will have the option to choose either the Laplace solver or the time-step solver. Where you have a choice, you should normally choose the Laplace solver as it will generally be faster and more accurate. The Laplace solver also handles differences in scale within a problem better (e.g. large amounts in one compartment and small amounts in another).

Figure 7.2-1
The Calculate
Dialog



The Calculate dialog has check boxes that the user can use to enable/disable the checking of negative transfer rates or source fluxes. As negative rates and fluxes are not normally expected this can be a useful check, for example it might reveal an incorrect sign in a source or transfer expression. A maximum of three negative values will be reported during a calculation (it is possible that hundreds could occur in a probabilistic case). However very small negative values can be generated by the solver (e.g. 10^{-15}) where zero might have been expected. If small negative values are a problem you can remove them using the max function, for example replacing x with $\max(x, 0)$.

If at least one sampled parameter is defined (see *Sections 6.4* and *7.5*), the 'samples' and 'seed' fields will be enabled. Enter in the 'samples' field an integer in the range 1 to 32,767 (note that the time taken to solve the problem will be approximately proportional to the number of samples). The 'seed' is the base seed used by the sampling random number generator (from which the seed of each sampled parameter is offset). The offset to the seed is constant for

the lifetime of a sampled parameter. This increases the independence of the sampled parameters and makes probabilistic calculations relatively robust to the deletion of old parameters or the addition of new parameters. You might wish to change the seed to see how this affects the results. The seed is an integer in the range 1 to 999999999.

Sampling is carried out using a Monte Carlo scheme. The Calculation Dialog has an extra input for Monte Carlo calculations. This gives the first sample to be run. By default this is zero (AMBER samples are numbered from 0). To run a single sample, the first sample number can be set to indicate which sample and the number of samples set to one. It is also possible to run a block of samples starting at a particular sample number.

The 'do not calculate' check box is provided to allow sample information to be set before a calculation is run. Once the number of samples and seed is set, graphs and reports for sampled input parameters can be produced.

In order to help in finding which sample number is producing results that are interesting to look at separately, there is a built-in parameter, SampleNumber. This takes a value equal to the sample number. It can then be used in scatter plots to pick out the sample of interest. Note that SampleNumber is set to the "global" sample number, so if 10 samples are run starting from 100 the SampleNumber will equal 100, 101, ... 109.

Before a calculation is allowed to run, the following criteria must be satisfied:

- ▲ the number of compartments must be greater than zero;
- ▲ the number of contaminants must be greater than zero;

- ▲ the number of result times must be greater than zero;
- ▲ all the required parameters must be correctly defined;
- ▲ the start time must be less than the first result time.

The Progress dialog will be displayed while the calculation is taking place, and it reports the sample number, the current calculation time and the estimated percentage completed.

A calculation can be cancelled while it is running by pressing the 'Cancel' button on the Progress dialog. If it is a deterministic calculation any results calculated so far will be lost. If it is a probabilistic calculation any samples already completed will be kept. For example, if a 100 sample calculation is cancelled after 50 samples it will be the same as if a 50 sample calculation had been run to completion.

7.3 Results

The results of a calculation are stored in a binary file with extension '.adf' (AMBER data file) and the same stem as the case file. If a case file expects to have a binary file and one cannot be located, for example because the binary file has been renamed or removed, a warning message will be generated when the case is opened. An '.adf' file can be regenerated by running a calculation again.

During calculation temporary binary files are produced. These files are prefixed by the letters 'ab' (AMBER binary). These are cleared up automatically and should not be deleted while AMBER is running.

7.3.1 Report File

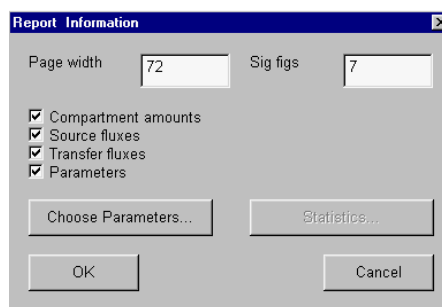
After the calculation the results can be presented in report form. If *Results | Report* is selected from the menu bar or the Report icon is selected from the toolbar, the Report Information dialog (see *Figure 7.3-1*) will appear.

The Report Information dialog gives the opportunity to decide the page width and number of significant figures to which the results are given in the report. The user has the choice of whether the compartment amounts, source fluxes, transfer fluxes and parameters are reported. The user can also define the units of time used for result times. If parameters are selected you can choose which parameters to output, and their ordering, by clicking on the 'Choose Parameters...' button. This will bring up the Choose Parameters dialog. If you only wish to report the results for a few of the available parameters, first use the double left arrow to move all the parameters from the Chosen to the Not Chosen list. Then choose an individual parameter by selecting it from the list of Not Chosen parameters and clicking on the single right arrow to move it into the Chosen parameters list. To select several parameters at once, hold down <control> and select the required parameters. To select a block of parameters from the list, hold down <shift> and select the required parameters.

Double-clicking on a parameter can also be used to move it from one list to another.

Figure 7.3-1

The Report
Information
Dialog



If the calculation was probabilistic (see *Sections 6* and *7.2*) you can choose which statistical measures to output by selecting the 'Statistics ...' button.

Once these options have been chosen the 'OK' button should be clicked and a dialog will appear prompting the user to supply the path for the file.

The report is written to a text file which can be read with an appropriate editing package such as Notepad in Windows.

Note that a probabilistic calculation with only one sample is treated like a deterministic calculation for the purposes of report generation.

Problems may be reported for some parameters that were not reported during calculation. This is because these parameters did not need to be evaluated during calculation. In such cases an error dialog will appear on the screen providing the user with information necessary to identify the cause of the evaluation error.

7.3.2 Graphs

You can create a graph by selecting *Results | Graph* from the menu bar or the Graph icon from the toolbar. The Graph type dialog will appear (see *Figure 7.3-2*).

If the calculation was deterministic the allowed graph types are:

▲ Line.

If the calculation was probabilistic (see *Sections 6* and *7.2*) the allowed graph types are:

▲ Line;

▲ Mean + Confidence Limits;

▲ Cumulative distribution function;

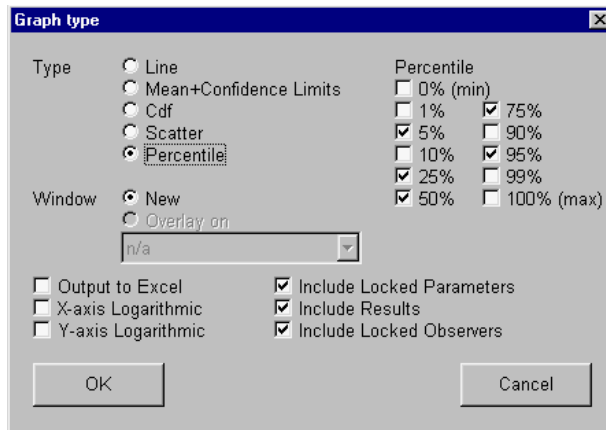
▲ Scatter;

▲ Percentile (see *Section 7.3.2.2*)

If you choose 'Mean + Confidence Limits' there is also a choice of bounds.

Note that graphing of input parameters is also allowed before a calculation.

Figure 7.3-2
The Graph Type
Dialog

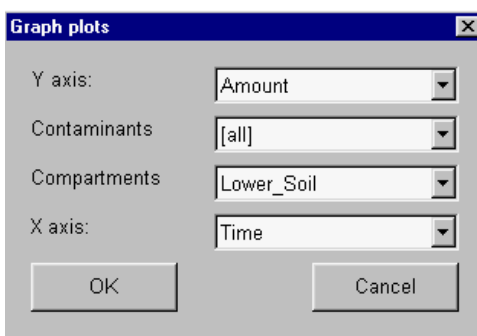


The axes types (log or linear) can be specified in the graph type dialog (see *Figure 7.3-2*), and are remembered between graphs. Thus, if a series of graphs are created which all require a logarithmic time axis, this can be achieved directly rather than by editing each one.

Depending on the type of graph chosen you will be prompted with one or more dialogs asking you for the parameter(s) you wish to plot (for example see *Figure 7.3-3*).

If a zero divide occurred in the evaluation of a parameter being graphed, the graph is not shown and an error message is reported.

Figure 7.3-3
The Graph Plots
Dialog



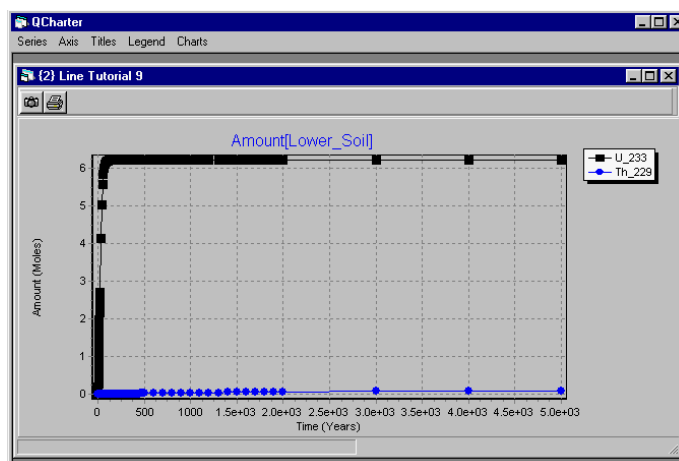
7.3.2.1 Charting Package

All graphing in AMBER is controlled through a charting package called QCharter, which is included with the AMBER software. This can produce charts itself and/or provide a link to Excel®. The chart data can also be tabulated.

The QCharter charting capabilities are described here; the Excel linkage is described later.

When the first chart is requested in AMBER, the QCharter package is launched (note that this gives a short delay while the component is loaded). This provides a window to which individual charts are added. A menu and toolbar provide various facilities for modifying the look of the charts, printing them or displaying data. An example showing a single chart is displayed below (see *Figure 7.3-4*).

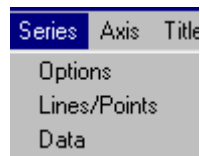
Figure 7.3-4
Example Chart



The controls are generally self explanatory, but a summary is given below.

The Series menu has three sub items.

Figure 7.3-5
Series Menu



Options allows lines to be switched on or off, and controls the series titles.

Lines/Points allows control over line and point colours, patterns, sizes, etc.

Data causes a tabulated view of the chart data to be displayed.

The axis menu has two sub items.

Figure 7.3-6

Axis Menu

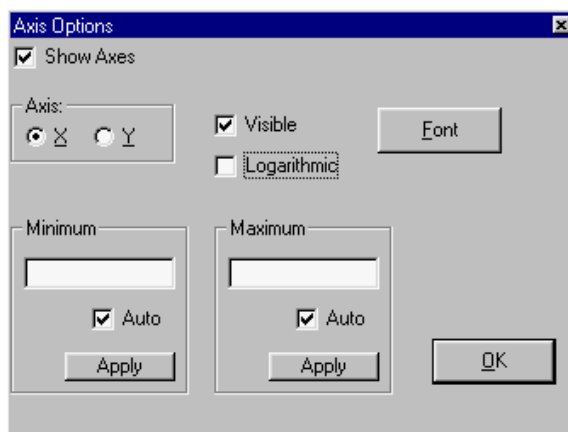


Show allows the axes to be displayed or hidden.

Options brings up a dialog for controlling the axes.

Figure 7.3-7

Axis Options Control Dialog



This controls both axes. Note that the minimum and maximum values only take effect when the corresponding Apply button is pressed. This is to avoid problems with illegal ranges and negatives for log axes. The Font button controls the font of the labels.

The Titles menu allows all titles to be set.

The Legend menu controls the legend position, and allows it to be hidden.

Charts gives a list of current charts, and allows tiling, or cascading of the chart windows.

The two toolbar items, the camera and printer, allow copying to various formats and printing.

On the chart itself, clicking on a point causes its coordinate to appear in the status bar at the bottom. Note that this can be done with point markers not shown, but is much easier when the markers are displayed.

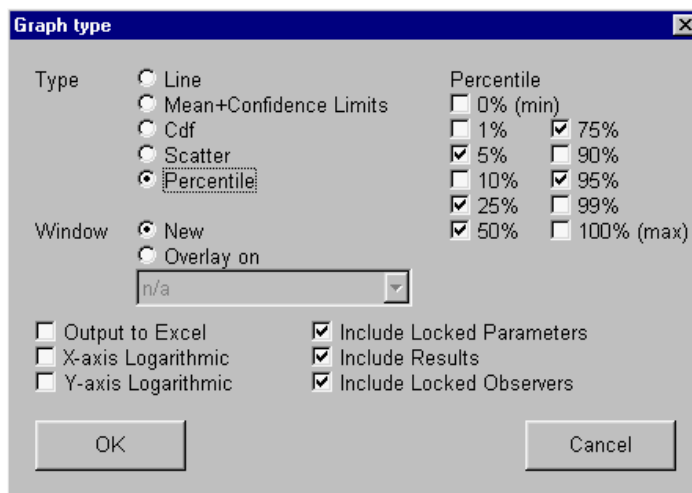
AMBER will add new charts to the same window, and allows charts to be overlain onto existing ones. Closing charts or the whole QCharter window while AMBER is running will not cause any problems – a new QCharter will be started up if another chart is plotted.

7.3.2.2 Percentile Charts

A percentile chart type has been made available for presenting the results of probabilistic calculations. Percentile plots draw user-selected percentiles for a time-dependent result. The chart type dialog offers a choice of 0%, 1%, 5%, 10%, 25%, 50%, 75%, 90%, 95%, 99%, and 100%. The 0% and 100% are used to show the smallest and largest values seen. The others are estimates of the percentiles from the sampled results. The chart type dialog is shown below (see *Figure 7.3-8*).

Figure 7.3-8

Chart Type
Dialog



7.3.2.3 Excel Linkage

The output of charts to Excel is routed through QCharter, which creates a new Excel workbook and add sheets and charts to it.

To output a chart to Excel, simply check the “Output to Excel” check box on the chart type dialog. If Excel is not available, then the internal charting package is used.

It is also possible to overlay new charts on the previously created Excel charts.

Each chart has an associated data sheet. Note that editing the names of the sheets should be avoided while AMBER is running. Closing Excel will simply cause a new one to be opened if further graphs are drawn.

7.3.3 Sensitivity File

As discussed in *Section 6*, AMBER can produce files for input into the external statistic package, SPSS. Sensitivity files are generated by selecting *Results | Sensitivity*. This creates a header file with the extension '.hdr' and a sensitivity file with the extension '.sns'. The default file stem is the same as for the case file. The header file gives details of the information contained in the sensitivity file.

The format of the '.hdr' file is:

```
SAMPLED: <list of sampled parameters>

SAMPLES: <number of samples in the
calculation>

PARAMETER: <the parameter being checked>

Contaminant: <the contaminant being checked>

Compartment: <the compartment being checked>

TIMES: <list of result times>
```

The format of the '.sns' file for p sampled parameters, s samples and r result times is:

```
<value of sampled parameter 1 for sample 1>
... <value of sampled parameter p for sample
1> <value of chosen parameter for given
compartment/contaminant for result time 1 and
sample 1> ... <value of chosen parameter for
given compartment/contaminant for result time
r and sample 1 > ...

<value of sampled parameter 1 for sample s>
... <value of sampled parameter p for sample
s> <value of chosen parameter for given
compartment/contaminant for result time 1 and
```

```
sample s > ... <value of chosen parameter for  
given compartment/contaminant for result time  
r and sample s>
```

This information can be used (for example by other packages) to check the sensitivity of a parameter to the values sampled.

7.3.4 Dump File

You can generate a dump file by selecting *Results | Dump*. This creates a text file containing comprehensive details of the results of a calculation on a sample by sample basis. This file can be very large, especially for probabilistic calculations.

Section**8**

8. Additional Features

8.1 Model Window Basics

The model window layout is controlled by the user through a combination of menus (including pop-up) and mouse actions. In general, a change is applied to all selected items through the main View menu, or to an individual item through a pop-up menu activated by the right mouse button. Some changes are global (e.g. the base font, line colour and shadow colour) and so do not rely on which items are selected.

8.1.1 Splitting Transfers

For complex AMBER cases, the structure of the model system can become obscured by the lines representing transfers between compartments. It is often sufficient, when considering a sub-system, to be visually aware that there are incoming or outgoing transfers without having to immediately see where they connect.

In order to allow this split transfers have been introduced. A split transfer is a transfer, which is visually represented in two parts, one

connected to the donor compartment and one connected to the receptor compartment.

The transfer itself is unaffected - only its visual representation changes.

Splitting a transfer is achieved by using the right mouse button and selecting Split on the pop-up menu. Alternatively, the transfer can be selected and the “s” key pressed on the keyboard.

Rejoining a split transfer is achieved by selecting Rejoin from the pop-up menu or “j” from the keyboard. A rejoin can be initiated from either half of the transfer – the rejoined transfer will use the position and attributes of the chosen half.

Once a transfer is split, its two halves are labelled using the transfer name with three dots (‘...’) as a suffix (for the half from the donor) or prefix (for the half to the receptor).

The two halves of a split transfer act completely independently as far as the view is concerned. Transfer information can be edited from either. Either can be used to delete the transfer.

8.1.2 Selection

Selecting a set of items (compartments, transfers and sources) is the first step in many layout changes. There are several ways of making and changing selections. Selection changes (in common with all layout changes) should be undertaken with the arrow tool.

The selected items are shown with white text on a black background. The more conventional approach of colour reversal has not been used because the user can set the colours of different items, making the selected ones less clearly visible.

The most basic selection method is by directly clicking on a single item. This will deselect all other items and select the particular one. By holding the “shift” key while clicking, no de-selection occurs, so the particular item is added to the selection. If the item clicked on when the “shift” key is held down is already selected, then it is removed from the selection.

To select a group of items, the mouse can be used to drag out a bounding rectangle starting at a position without a source, compartment or transfer. All items with centres in the rectangle will be selected and items outside the rectangle will be de-selected. Note that dragging with the “shift” key held is not currently supported.

The *View | Select* menu allows all of a particular type of item to be selected. This selection is added to any existing selection.

The *View | Deselect* menu allows all of a particular type of item to be removed from the selection.

According to which items are already selected, the menu items under *View | Select* and *View | Deselect* may be greyed out to indicate that a particular action is inapplicable.

By combining the use of the mouse-drag selection and the menu de-selection option, it is possible to select all of a particular type in a section of the model.

Using the right mouse button on a particular item enables it to be changed without any change to the selection.

8.1.3 Zoom

A useful feature for cases where the size of the model in the Model window needs to be modified is the zoom function. The Model window can be scaled to 50%, 75%, 100%, 150%, 200% or any

percentage between 10 and 500 (through *View | Zoom | Other*). The zoom function is accessed through the zoom sub-menu in the View menu, i.e. *View | Zoom*.

The area shown in the Model window may be changed by using the scrolling bars at the right, and bottom edges of the window, or by using the arrow keys.

Occasionally, particularly after zooming, parts of the model view may become hidden under the palette. Shifting the view in the Model window using the arrow keys may easily retrieve the view. Three key-combinations can be used. Shift-LeftArrow, moves the whole layout to near the left boundary; Shift-UpArrow moves it to near the top; Shift-Home does both together.

Allows the user to view the model in different orders of magnification. The Model window can be zoomed to a user-specified magnification. Views and fonts are scaled together.

8.1.4 Colour

Colours can be set for every separate compartment, transfer and source in the system. The colour of all lines is the same and can be set, as can the colour of shadows and the background colour.

To set the colour of a selected set of items, the *View | Colour* menu should be used. To set the colour of an individual item, the pop-up menu Colour option should be selected. The pop-up menu can be used to set the background colour by using the right mouse button away from any model items.

When the colour menu item is selected, a colour dialog box will appear. This is the standard Windows dialog. Where possible the current colour will be selected initially. Custom colours can be used, and will be saved in the AMBER case file.

For compartments, transfers and sources, the chosen colour is the background colour of the text box. The text itself will be black or white depending on how dark the selected colour is.

8.1.5 Sizes

The relative size can be set for every separate compartment, transfer and source in the system. This is relative to the nominal standard size. The text for a resized item will use a scaled version of the base font.

Sizing should not be confused with zooming. When a zoom factor is set, the whole window is drawn with this factor – so that all apparent sizes and positions change. When a relative size is set, the positions are not affected.

To set the relative size of a selected set of items, the *View | Relative Size* menu should be used. To set the relative size of an individual item, the pop-up menu Relative Size option should be selected.

The relative size is chosen from the given list. When the pop-up menu is used the current relative size of the particular item will be indicated – this can be used to discover the current relative size. When the main menu is used, the relative sizes of all selected items will be indicated.

8.1.6 Shadows

The drop shadows can have their colour changed through the *View | Colour* menu or may be hidden completely through the *View | Hide Shadows* menu item. This is a checkable menu item, so a tick means that the shadows are hidden. Each time the menu item is selected the choice is toggled between hidden and shown.

8.1.7 Layout Management

Every AMBER case has an active layout, corresponding to what is seen on the screen. This is saved in the case file and re-used when the case file is opened subsequently. At any time, the active layout can be stored under a user-chosen name. Stored layouts are also saved in the case file. Stored layouts can be used to replace the active layout at any time. There is no limit to the number of layouts that can be stored.

It is permitted to add new model items after layouts have been stored.

All layouts are initially created as the active layout and the saved under a user-chosen name.

All the layout management functions are accessed through the *View | Layout* menu. This has three items: Use, Store and Delete.

The *View | Layout | Use* menu item allows a stored layout to be used as the active layout. It also allows the default colour, size and font settings to be applied. If the active layout that is being replaced has not been stored then a warning will be given. If there are items in the model that did not exist when the stored layout was created then these will be placed near the top left of the model window, using default colours and sizes.

The *View | Layout | Store* menu item allows the active layout to be stored. This can replace an existing stored layout or create a new one. Note that it is necessary to use Store to replace a stored layout if any required changes are made in the active layout. Changes to the active layout are never automatically made to a stored layout.

The *View | Layout | Delete* menu item allows the stored layouts to be removed. Note that there is no rename function – but the same

effect can be achieved by making a layout active, storing it under the new name and then deleting the old one.

When the active layout has been set using a stored layout, the layout name appears on the title bar of the model window. If the layout is changed from the stored version, an asterisk is inserted after the layout name.

8.1.8 Free Text on Model Windows

The model window and submodel windows can now have short text items added to them. The bottom palette tool should be used. Clicking with it adds a free text item. Double-click on this to edit the text. The colour and font can be changed for each free text item separately, using the right mouse button. If no font has been specifically given for a free text item it will use the chosen window font.

Free text is intended for labels, so multi-line capability has not been provided.

The free text items have a transparent background so that they can be placed on top of other items if required.

8.2 Submodel Hierarchies

Submodels involve splitting a complex model into convenient groups to aid presentation and accessibility.

A submodel is simply a group of compartments (and sources) that form a convenient unit. Submodels can be nested – so a submodel can contain other submodels.

Transfers do not belong to submodels directly. If both ends of a transfer are in the same submodel then that transfer will only be visible in that submodel. Transfers between compartments in different submodels will be visible in several submodels, as described in more detail later.

8.2.1 Creating Submodels

To create a new submodel, the submodel tool should be selected from the palette on the model window (which has six items in it, the fifth being the submodel tool). An empty submodel can be created by clicking or dragging out an empty rectangle with the tool. If existing compartments and sources are to be put into the new submodel, then a rectangle containing them should be dragged out.

Double-clicking on a submodel allows the name and description to be changed.

Where transfers or sources enter or leave submodels, this is denoted by a small arrow entering or leaving respectively. These arrows show entry or exit from above, because the parent submodel is conceptually at a higher level. For clarity, it is recommended that such transfers be placed above the items that they connect to within a submodel.

8.2.2 Submodel Windows

Each submodel has its own window, which looks exactly like the top-level model window. When a new submodel is made, its window is opened. Otherwise, submodel windows can be accessed in a number of ways. The Windows menu has a Submodels submenu that allows direct access to each window. Alternatively, the hierarchy can be followed up and down.

To go to a submodel window, use the right mouse button on a submodel and select the Submodel Window item in the pop-up menu.

To go back to the parent submodel, use the right mouse button on the background and select Parent Window.

Several submodel windows can be open at one time. Selecting an already open window brings it to the front. If changes to submodel structures are being made, it is best to avoid having too many submodel windows open, since each has to be updated when a change is made.

8.2.3 Laying Out Submodels

The submodel box on the parent window can be resized freely by selecting it and using the grabs that appear in each corner. Because of this, the relative size function applies only to the font on a submodel box.

After a submodel is created, the layout can be improved with the help of two items available on the pop-up menu. Firstly, all transfers into and out of a submodel can be split (Split All). Secondly, the transfer positions can be tied to the submodel (Move Transfers), so that moving the submodel takes all the transfers with it.

Colour-coding the submodel boxes (and perhaps the background of the submodel window to match) is recommended.

When a new submodel is created with initial contents, the position of the items in the window, and the style, is taken from their previous positions. If items are added individually by drag and drop, their positions will be determined by the drop position, but their style will revert to default. For any split transfers that become

relevant for a submodel, the label moves with the submodel or compartment. For connected transfers, the label is not moved.

Storing and reusing layouts is currently not supported for submodels.

Note that if systems with large numbers of compartments are being modelled, the creation of new windows can be relatively slow. This is because of the checking needed to determine which transfers are to be shown.

Note that all submodels and compartments have “Join All” and “Split All” on their pop-up menus, available through the right mouse button.

8.2.4 Editing Submodels

Clicking the right mouse button on the background of a submodel now gives access to the Submodel Edit dialog, so that a submodel can be renamed from its own window.

The F8 key can be used to switch to the parent submodel of the submodel being viewed.

New items can be added into submodels just like adding them to the model window. Items can also be deleted from submodels in the same way. Items can also be moved between submodels as follows.

Open the windows for the submodel containing the item (compartment, source or other submodel), and the window where you want to move it. Now use the control (Ctrl) key and the mouse together to select and drag the item from one window to the other. The item will be added where it is dropped. Dropping it in the same window has no effect.

Adding a transfer between compartments in different submodels must be done in the common parent window. Add the transfer from one submodel to the other, and select the appropriate compartments from the list that is presented. This also works for sources. Of course, the position of the transfers in the submodels will need to be changed manually after the transfer is added. Note that transfers within submodels can be added by creating a transfer to and from the same submodel in the parent window, but this is not recommended.

Editing and deleting transfers and sources can be done from any submodel in which they appear.

Where a transfer is between compartments that are not in the same submodel, the transfer will be visible in the submodels that contain its donor and receptor and in submodels containing these, up to a common parent. Transfers from the donor are indicated by an out-pointing arrow, and by the name having a “...” suffix. Transfers to the receptor have an incoming arrow and a “...” prefix. The switch between the two styles occurs at the common parent.

8.2.5 Submodel Deletion

Submodels can be deleted in the same way as any other item, except the user is also asked whether the contents should be deleted or kept.

Deleting the contents deletes all compartments and nested submodels within the deleted submodel. Keeping the contents removes the submodel and moves its contents into the parent submodel.

The new layout is adjusted so that these are visible and sensibly placed.

8.3 Availability Schemes

In older versions of AMBER, no non-linear behaviour was allowed. Specifically, the transfer-flux between compartments was restricted to being linearly proportional to the amount of material in the donor compartment. That is, the transfer flux was defined by the relationship

$$\text{TransferFlux} = \text{TransferRate} * \text{Amount}[\text{donor}].$$

Two particular types of non-linear behaviour have been introduced that is sufficient to cover the common cases of solubility limitation and sorption.

8.3.1 The Concept of Availability

Generally, the reason for a non-linear relationship between the transfer flux and the amount of contaminant in the donor compartment is that only a fraction of the contaminant is available for transport. In the case of solubility limitation, excess material will be present as precipitate and so will not move in the aqueous phase. Here, we introduce the general concept of Availability to handle these situations.

The availability of a contaminant for a particular transfer is defined to be the fraction of that contaminant that participates in the transfer. Thus, availability is between zero and one.

By default, the availability of all contaminants for all transfers is unity. By selecting from a set of options, the user can specify a prescription for determining the availability for each contaminant for a nominated transfer. Note that an Availability is attached to a transfer rather than a compartment. This is because different transfers might have different availabilities for the same

compartment (e.g. if one acts through the aqueous phase and the other on the total, as in erosion).

After a calculation, Availability is a built-in parameter and the formula for transfer flux is now

$$\text{TransferFlux} = \text{TransferRate} * \text{Availability} * \text{Amount}[\text{donor}].$$

Because Availability depends on the Amount, the equations that AMBER solves become non-linear. This therefore forces the use of the time-stepping solver (Dylan). The following sections discuss how the user prescribes the scheme to be used for calculating Availability in a particular transfer. For each scheme, the user must consider what information needs to be provided, and how this is used.

8.3.2 Individual Limit Scheme

For an individual limit scheme, all that needs to be specified by the user is the maximum amount for each contaminant that should be included in the transfer. This corresponds to the amount in the compartment when the solubility limit is reached, and could be written in this way, for example

$$\text{Limit} = \text{Volume} * \text{Porosity} * \text{RetentionFactor} * \text{SolLimit}.$$

Given this limit, the Availability is the minimum of Limit/Amount and unity. It is an error for the Limit to be negative or zero. If the Amount is zero or negative, the Availability is defined to be unity.

The user does not directly specify the formula for Availability – this is done automatically by AMBER. Rather, the user specifies the parameters needed to calculate Availability.

The Limit can be time-dependent and/or sampled as needed by the user. If the value is specified as a constant it should explicitly

have units specified – e.g. 7[mol]. See the discussion in *Section 4.3.4* about units.

8.3.3 Shared Limit Scheme

For a shared limit scheme, in addition to the limits it is necessary to define how the contaminants group together. For cases where this is necessary, the user must have defined a NameSet for this purpose (typically Elements) and a mapping from Contaminants to this NameSet (e.g. element).

The information needed here is the mapping (which defines the target NameSet and the assignment of contaminants) and the Limit for each item in the target NameSet. For the specific case of elemental solubility limits, the Limit would be defined for each element and the mapping would indicate which contaminant belongs to which element.

The Availability is defined for each contaminant, but the values for contaminants mapped to the same target (e.g. isotopes of a particular element) are all equal. This ensures that the available amounts are in proportion to the full amounts (i.e. that the correct isotopic fractions are preserved). See the note about units for further discussion.

The Availability is the minimum of Limit/SummedAmount and unity. It is an error for the Limit to be zero or negative. The SummedAmount is the Amount summed over contaminants mapped to the same target. If the SummedAmount is zero or negative, the availability is defined to be unity.

The Limit can be time-dependent and/or sampled as needed by the user.

See the discussion about units in *Section 4.3*.

8.3.4 Langmuir Scheme

In some circumstances the relationship between the total amount of a contaminant and the amount available for transport is governed by a non-linear sorption relationship. Typically, there is a bound on the amount that can be sorbed, and so the fraction available for transport increases with the total amount. This is in contrast to a solubility limit scheme where the availability reduces as the amount increases.

A common sorption isotherm is the Langmuir isotherm. When this applies, the availability can be written as a simple function of the amount.

$$Availability = \frac{Amount + \alpha}{Amount + \beta}$$

where the values of α and β are determined by the precise details of the model.

Two cases need to be considered, as with the limit case:

- ▲ an individual scheme, where each contaminant is independent, and
- ▲ a shared scheme, where the total over several contaminants controls the availability.

Both cases are available. For the shared scheme, a mapping must be supplied that groups the contaminants in the correct way, and the parameters will then depend on the target of this mapping (typically Elements).

The graphical interface asks for two expressions, one for the top (α above), and one for the bottom (β above).

Note that if availability is required for some contaminants only, then α and β values for the others should be set equal to one another. Any value is valid, but zero should be avoided because of the possibility of the denominator going to zero.

See the discussion on units in *Section 4.3*.

8.3.5 Availability Dialog

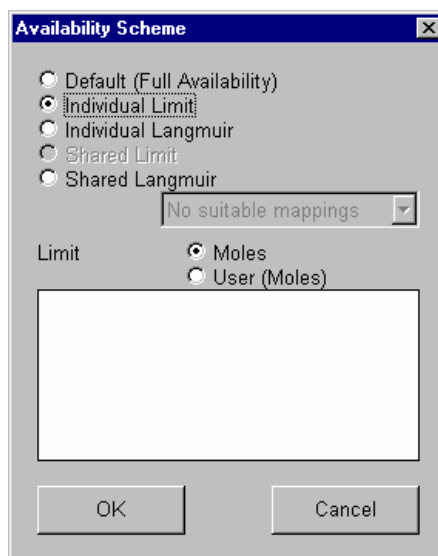
The extra user-interface components are:

- ▲ A way of indicating that a non-default Availability Scheme is to be used for a particular transfer.
- ▲ A way of specifying the parameters needed by the Availability Scheme.

A button on the Transfer Dialog gives access to a new Availability Dialog shown below.

Figure 8.3-1

Availability
Dialog



The drop-down list box shows valid mappings (i.e. those from Contaminants). The expression given under Limit must give a value for each of the target NameSet, or for each contaminant in the case of Individual Limits. The user can indicate the units that are being used, see the discussion in *Section 4.3*.

8.3.6 Scaling in the Solver

When availabilities are used, particularly to represent solubility limitation, the amount of a contaminant that moves from its initial compartment can be very small. This could lead to problems with relative errors causing the solver to become inefficient.

To avoid these problems, the amounts used to calculate the relative errors are scaled by the availability for the outgoing transfers.

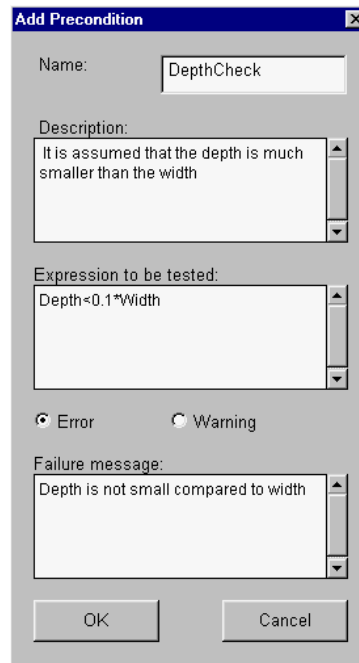
8.4 Preconditions

When a complex model is set up in AMBER, there may be assumptions made about the range of parameter values for which it will be used. These can be checked and because the checks apply to model inputs, they can be tested before running. They are known as preconditions.

Preconditions are added through a Preconditions window and dialog. The window is accessed from the Windows menu (there is no tool-bar item). This gives a list of preconditions, which can be modified as needed. Adding a new precondition or editing an existing one (by double-clicking) brings up the preconditions dialog.

A precondition has a name, a description, a test and a failure message. The name and test are compulsory. The name must satisfy the usual rules for a valid name in AMBER (starts with a letter, no spaces or special characters).

Figure 8.4-1
 Edit
 Preconditions
 Dialog



The test is an expression that gives a true or false result, such as $\text{param1} > 0$. Since AMBER has no special boolean type, the value zero corresponds to false, and a non-zero value to true. See *Section 5.5* for a list of the operators and functions that produce boolean results.

Preconditions can be set to give errors or warnings. To stop a calculation going ahead if the condition fails, set the type to error.

All preconditions can be checked by using the Check Preconditions Window, accessed from the Results menu. This lists failures, and flags errors and warnings. The user-specified failure message is written out for each failure. The preconditions are also checked automatically prior to calculation, but here only errors are

relevant and the checking stops if any violated condition is discovered. If a precondition contains an error (e.g. refers to a non-existent parameter) then this will also count as an error.

Note that preconditions can be time-dependent, but the checks will only be made at result times.

Preconditions can have multiplicity – the check will be made for each entry.

The dialog box shown above (*see Figure 7.4-1*) gives an example of a valid precondition.

8.5 Command Line Arguments

You can specify the name of a file to open on the command line. If the file is not in the same directory as the AMBER executable file you will have to specify the path.

Command line arguments can be used to specify one or more cases to be read at start-up. Choose *Start | Run* and enter the path to the AMBER executable followed by a space and the name of the cases to be loaded e.g.

```
c:\AMBER4\AMBER.exe Tutprob.cse
```

You can also use *File | Associate* from the file manager to associate AMBER with all ‘.cse’ files. This will allow you to start AMBER and load a case file by just double clicking on the ‘.cse’ file in the file manager.

If the ‘.cse’ file is in a different directory to the AMBER ‘.irs’ string resource files you will need to set the ‘RESOURCE_PATH’ environment variable (*see ‘Starting AMBER’ in Section 3*).

Section**9**

9. Trouble Shooting and Feedback

9.1 On-line Help

This reference manual has been made available as on-line help. This uses the standard Windows help system. It can be launched from the Help menu.

In this version, the help is not context sensitive, this will be added in later versions. A comprehensive set of keywords has been added, so using the search facility should quickly find any information required.

9.2 Errors at startup

If any errors are detected before the graphical user interface starts up they will be logged in the file 'errors.txt'.

9.3 Parameters

Setting up a complex model within AMBER may require the input of a large number of parameters, each of which must be correctly defined in order to allow the model to function properly. Some errors in parameter input may be obvious as they are brought to the user's attention by AMBER error messages when the model fails at the calculation or report generation stage but some errors may go unnoticed until the final output of the model is received. These problems, and how to resolve them, are discussed below.

9.3.1 Combining Parameters: the Need for Care with Cases

A typical compartmental model may contain in excess of fifty different parameters, all of which must be correctly defined. Frequently several parameters may be defined by referencing other input parameters, for example, assume the 'depth of recharge' parameter 'D_rec' is defined in terms of the equation:

$$D_rec = Ppt - Etp + D_irr * (1 - mu_root)$$

A common mistake which is made when setting up a model is to initially define a parameter in upper case alphabetical characters and then reference the same parameter in lower case (or vice versa), elsewhere in the model. As case is significant in AMBER care should be taken to avoid inconsistencies in the definitions and in the subsequent referencing of input parameters. In the above example the evapo-transpiration rate 'Etp' has been correctly referenced, however if 'Etp' were to be replaced by 'ETP' then the model would fail at the calculation stage. The user would be informed of this error by an appropriate error message (see below).

9.3.2 Combining Parameters: the Need for Consistent Multiplicity

It is important to realise that a given parameter should only be defined in terms of other parameters that have with the same or lower degrees of multiplicity. For example, assume that the parameter 'D_w' has been defined as:

$$W_ing * D_ing * C_comp [Well]$$

'D_w' and 'D_ing' have contaminant multiplicity, 'W_ing' has no multiplicity, and although 'C_comp' has contaminant and compartment multiplicity, 'C_comp[Well]' only has contaminant multiplicity.

In order to check that parameters have been correctly defined the 'Check Parameters' function can be used (see *Section 6.1* for further guidance on this function).

9.4 Error Messages

It is recommended that before a calculation is run the 'Check Parameters' function is used. To do this select *Results | Check Parameters*. A dialog appears showing the number of parameters, and the number of errors. By using the scroll bars on this window the parameter definitions may be viewed. If there is an error in a parameter it will be highlighted by an error message. This error message will describe the type of error. Typical error messages are explained below.

'Evaluate error - cannot find definition for ETP [while evaluating D_rec] [while evaluating r_RC] [evaluating transfer [transfers = recharge 1]]'

This error message informs the user that no definition could be found for the parameter 'ETP' whilst AMBER was attempting to evaluate the parameters 'D_rec' and 'r_RC'. In order to rectify the problem the user would have to check the definitions of 'D_rec' and 'r_RC' as well as the definition of 'ETP'. From an inspection of these parameters it would become apparent that the evapo-transpiration rate was initially defined as 'Etp' and that in the equation defining the depth of recharge ('D_rec') a parameter 'ETP' has been referenced which has no definition. The rate coefficient for transfer of radionuclides by recharge 'r_Rc' is defined in terms of 'D_rec' and therefore cannot be evaluated either. To rectify this error the user must replace the parameter 'ETP' by 'Etp' in the equation defining depth of recharge.

Errors in matching the multiplicity of parameters are also highlighted at the calculation stage. For example:

```
'Evaluate error - Array dimension error [while evaluating  
r_Wirr] [evaluating transfer[*] [Transfers = Irrigation]]
```

This statement informs the user that during the evaluation of the parameter 'r_Wirr' there has been a mismatch in the degrees of multiplicity. In order to correct this error the user would have to examine the definition of 'r_Wirr', inspection would reveal that 'C_comp' (with contaminant and compartment multiplicity), rather than 'C_comp[Well]' (with contaminant multiplicity) has been used. Since 'r_Wirr' only has contaminant multiplicity, 'C_comp' cannot be used in its definition.

If an error has been made in the definition of an observer parameter e.g. in total dose ('D_tot') then the calculation will proceed without notifying the user of this error.

Observer parameters are calculated in a post processing step after the solver has completed the initial calculation. Once the case has been calculated it is therefore important to use the 'Check Parameters' function. This will now also check for errors in Observer Parameters. The case can not be fully reported until these errors have been corrected.

9.5 Common Syntax Errors

Spaces are not permitted in parameter names therefore the underscore symbol '_' is frequently used in AMBER. Care should be taken to avoid using a minus sign '-' in place of the underscore.

The symbol for multiplication is '*' and not 'x'.

9.6 Copy and Paste

As with most Windows applications, considerable time can be saved by using the copy and paste functions.

Copy: Highlight text to be copied then press

<CTRL> and <c>

Paste: Press <CTRL> and <v>

This will also allow copying and pasting between AMBER and other Windows applications and between AMBER models.

9.7 Choosing Result Times

It is important to realise that the result times chosen can vastly affect the appearance of any graphs plotted, although the final results will not be changed.

Care must be taken to ensure that appropriate result times are chosen.

9.8 Speeding Up The Time-step Solver

When the time-step solver is being used AMBER may slow down considerably. This can be due to AMBER calculations running at a higher accuracy than required. This can normally be overcome by putting a start amount in a compartment. The user is however reminded to make sure this starting value is of the same order as that of the results, otherwise errors may be significant.

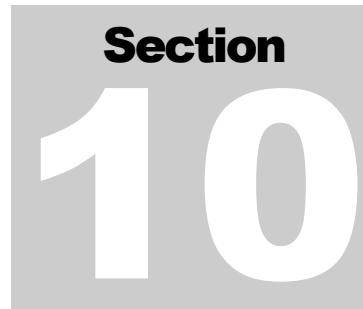
Despite this adjustment the time-step solver may still be slower than the Laplace solver.

In AMBER 4.5 the internal editable parameter `MinChangeTime`, can also be used to significantly decrease the run-time of case files.

See *Section 6.2.2.2* for further information.

9.9 Feedback

Please send comments and details of problems, clearly marked “AMBER 4.5”, to the contact identified in the ‘readme.txt’ file.

A grey square graphic with the word "Section" in a bold, black, sans-serif font at the top. Below it, the number "10" is written in a very large, bold, white, sans-serif font.

10. Restricted Versions of AMBER

A restricted version of AMBER can now be produced. This is intended to be used in delivering a specific solution to a user who does not want to purchase a full AMBER licence.

The restricted version has to be used in conjunction with a permit file. The permit file contains a set of conditions that must be satisfied by a case for it to be valid. Currently this is in terms of the number and names of compartments, contaminants etc., but could be extended as required.

It is possible to restrict the usage strictly to the original model or to allow a limited degree of extendibility.

Permit files are encrypted from a text version that contains the conditions. They also contain the name and purpose of the permit. The encryption code is separate from AMBER. Any released version of AMBER can have an associated restricted version.

Index

A

Availability, 30, 138–43, 159
 individual limit, 139
 langmuir, 141
 shared limit, 140

B

Backward compatibility, 12
 Boolean expressions, 87, 159

C

Calculation, 16, 99, 109–14
 Compartment, 35, 36, 160
 Constants, 21
 Contaminant, 37, 38, 42–45, 92,
 113, 160

D

Decays, 44, 45–51, 160
 compartment dependent, 50, 162
 Derived observers. *See* Parameters
 Descriptions, 31
 Deterministic calculation, 116, 160
 Dongle device driver, 10
 Dongle protection, 10
 Dump file, 125

E

Errors, 85, 107, 116, 149, 150, 151,
 152
 array dimension, 152
 common syntax, 153
 Expressions, 84–91, 161

F

Free text, 133
 Functionality, 9

G

Glossary, 159
 Graphs, 117–23

I

Installation, 9
 Internal parameters, 20, 26, 43, 72–
 74

L

Laplace, 109, 111, 161
 Literals, 80–81, 162
 Locking parameters, 74, 162
 Lookup functions, 29, 89–91

M

Mappings, 81–84, 162
 Mathematical functions, 86, 87
 Menu, 14–19
 Model, 18–42
 Multiplicity, 56, 91–92, 151, 152,
 162

N

Names, 20
 NameSets, 20, 54, 77–80, 91, 162
 Negative results, 64

O

Observers. *See* Parameters
Operators, 85–91
Order of precedence, 87

P

Parameters, 51–74, 150, 152, 153, 163
 checking parameters, 107
 derived observers, 65–66, 160
 observers, 64, 153, 163
 sampled parameters, 62–64, 100, 101, 163
 standard parameters, 53, 164
 switch parameters, 60–62, 164
 time dependent. *See* Time dependent parameters
Preconditions, 144–46, 163
Printing, 15
Probabilistic calculations, 113, 116, 163
Probability density functions
 beta, 103
 gaussian, 102
 general CDF, 103
 triangular, 102
 truncated gaussian, 102
 uniform, 101

Q

QCharter, 119–22

R

Report, 16, 107, 114–16

Result times, 92–96
Results, 16, 108–14, 163

S

Sampled parameters. *See* Parameters
Sampling, 99–101
Saving, 96–97
Searching, 75
Sensitivity, 16, 99, 124–25
Snapshots, 95–96, 164
Solubility limitation. *See* Availability
Source term, 35, 42, 70, 164
Start amount, 37, 38, 164
Start time, 93, 114, 164
Submodel, 76, 133–37, 164
System requirements, 9

T

Time dependent parameters, 56–60, 165
 cyclic time-dependent, 58
 cyclic-time-dependent, 160
 piece-wise time dependent, 58
Time-step solver, 109, 111, 165
Toolbar, 8, 19
Transfer process, 165
Transfers, 36–42
 non-depleting, 40, 163
 splitting transfers, 127, 164

U

Units, 17, 21–31, 72, 93
 user defined units, 24

Appendix A.

Glossary of Terms

Active Layout - The layout in use for the model window at any particular time.

Alphabetic Character - A character that is one of A-Z or a-z.

Alphanumeric Character - A character that is one of A-Z, a-z or 0-9.

Availability - A fraction between 0 and 1 which is the amount of contaminant which can participate in a particular transfer.

Best Estimate - A single value used to represent a sampled parameter in a deterministic calculation. Usually, the best estimate will be central in the distribution, but this is not compulsory.

Binary Operator - An operator with two operands (e.g. the '+' in $X+Y$).

Boolean Operator - A nonmathematical operand which returns a value of 1 or 0 depending on whether the result is true or false.

Case - The entity that contains compartments, transfers, sources, contaminants, decays, parameters and result times. Cases can be stored to disk and read back.

Cdf - Cumulative distribution function.

Chain - A possible path through a series of decays. For example if the decays are $A \rightarrow B$, $A \rightarrow C$, $B \rightarrow D$ and $C \rightarrow D$, the possible chains are $A \rightarrow B \rightarrow D$ and $A \rightarrow C \rightarrow D$.

Compartment - A distinct physical entity (e.g. top-soil, river, cow or person) through which contaminants may pass and/or in which contaminants may accumulate. Instantaneous uniform mixing within a given compartment is assumed.

Contaminant - A substance (e.g. plutonium-242, mercury or lindane) which may be transported between compartments and may decay or degrade into other substances.

Cyclic Time-Dependent Parameter - A parameter which is piecewise-constant in time and goes through repeating cycles. Such parameters could be used, for instance, for parameters that depend on the time of year.

Decay Rate - An internal read-only parameter indexed over decays. It can be used in expressions and evaluates to the decay rate as specified in the decay dialog.

Decay Process - The process of transformation of one contaminant into another. Decays are assumed to be first order in nature (i.e. the rate of decay is proportional to the amount of substance remaining). 'Decay process' is often shortened to 'decay'.

Derived Observer - Derived Observers provide the mechanism to create an Observer by using the result at some particular time.

Deterministic Calculation - A calculation in which none of the parameters is sampled.

Dialog - A special type of window which the user has to close before any other task can be started. Dialogs are generally used to prompt the user for information.

Environment Variable - An operating system variable that can be created and set by the user.

Expression - A mathematical statement phrased in terms of parameters, constants, operators and functions that evaluates to a constant for any given time.

File Extension - The part of a file name after the ‘.’. For example the extension of ‘mycase.cse’ is ‘cse’.

File Stem - The part of the file name before the ‘.’. For example the file stem of ‘mycase.cse’ is ‘mycase’.

Flux - The amount of a given contaminant flowing moving between two compartments (in the case of a transfer) or into a compartment from outside the system (in the case of a source) per unit time.

Function - A procedure for calculating mathematical transformations e.g. ‘exp’ in ‘exp(x)’.

Laplace Solver - Solves Equation (1) by means of a Laplace transform and is therefore only valid when the source terms and/or transfer rates of the problem are constant or piecewise-constant functions of time.

Layout - A collection of information controlling the positions, sizes, colours and other attributes of the model window.

Library - A part of a case that can be saved independently and re-loaded.

Literal - A word in single quotes. Literals are useful in providing an indexing system over related quantities.

LocalDecayRate - An internal editable parameter indexed over Compartments and Decays. The default expression for LocalDecayRate is that it is equal to DecayRate in all compartments.

Locked Parameter - A locked parameter is protected from editing. It has been designated read only, and can not be edited until “unlocked.”

Mapping - A mapping links the elements of one NameSet to those of another. For example contaminants may be mapped to elements.

MinChangeTime - A timescale for the solver interpolation scheme.

Model - The model is a compartmental representation of a system.

Multiplicity - The ability of a parameter to have more than one expression/value. For example a Kd parameter might have contaminant multiplicity, allowing it to have a different expression for each contaminant. Only standard parameters, observer parameters (derived output parameters) and some internal parameters can have multiplicity. A parameter with contaminant multiplicity may alternatively be described as being ‘indexed’ by contaminant.

NameSet - A NameSet is a group of names. By defining NameSets the user can allow multiplicity by their own defined dimensions, and group similar names to simplify parameters and expressions.

Non-Depleting Transfer - A transfer process in which there is no reduction of contaminant in the donor compartment. This is in contrast to a depleting transfer where the amount received by the receptor compartment is matched by the amount lost from the donor compartment. Non-depleting transfer can be useful for modelling some physical processes.

Observer Parameter - A parameter that is calculated as a post-processing step after the solver has completed. Possible uses of observer parameters include the calculation of the intake of a contaminant by a human.

Parameter - A parameter is a named object with a defined value used in AMBER cases. There are several types of parameter-internal, standard, time-dependent, sampled, and observer.

Precondition - A check that is made prior to solving. This can be used to ensure that derived parameters stay in range (e.g. in a probabilistic calculation).

Probabilistic Calculation - A calculation in which at least one of the parameters is sampled.

Radio Buttons - A group of buttons on the user interface, only one of which can be selected at any time.

Result Time - A time at which results will be calculated.

Sample - A calculation for which each sampled parameter has been assigned a fixed value (by a random number generator). A number of samples will be required to evaluate probabilistic behaviour.

Sampled Parameter - A parameter that has a different value for each sample in a probabilistic run.

Scalar - A parameter with no multiplicity.

SnapShot Time - A time when the compartment amounts are stored in addition to result times.

Source Term - The input of one or more contaminants from outside the model into a compartment. 'Source term' is often shortened to 'source'.

Split Transfer - A transfer which is visually represented in two parts, one connected to the donor compartment and one connected to the receptor compartment.

Standard Parameter - The basic parameter type. It can have multiplicity and the expressions used to define it can reference any other type of parameter except 't' or observer parameters.

Start Amount - The amount of each contaminant in a compartment at the start time.

Start Time - The time at which a calculation begins.

Stored Layout - A named layout that has been saved and can later be used as the active layout.

Submodel - A group of compartments (and sources) that form a convenient unit and thus can be represented in a separate window.

Switch Parameter - A parameter which defines a time at which there is a discontinuity in the AMBER case must be defined as a switch parameter. Switch parameters alert AMBER to the discontinuity.

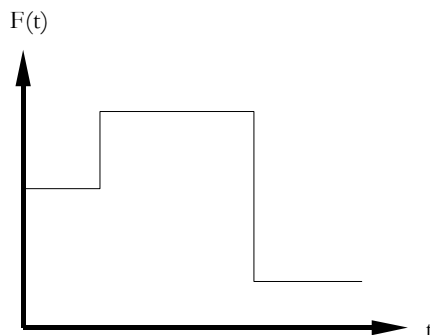
Switch Time - A time at which a time-dependent parameter changes its value.

Time-Step Solver - A 'stiff' ordinary differential equation solver that can be used to solve problems in which source terms and/or transfer rates are dependent on the variable 't' (time). It can also be used for problems in which source terms and/or transfer rates are constant or piecewise-constant functions of time.

Time-Dependent Parameter - A parameter that changes its value with time in a piecewise constant manner (see *Figure 5-1*).

Figure A-1

A piecewise constant function of time



Transfer Process - The transfer of one or more contaminants from a donor compartment to a receptor compartment. The flux of the transfer is equal to the amount in the donor compartment multiplied by the rate of the transfer. For example, if the transfer rate is $0.1y^{-1}$ then one tenth of the amount in the donor compartment will be added to the receptor compartment each year. 'Transfer process' is often shortened to 'transfer'. See also **non-depleting transfer**.

Unary Operator - An operator with one operand (e.g. the '-' in '-X').