R-11-01

Pandora – a simulation tool for safety assessments

Technical description and user's guide

Per-Anders Ekström, Facilia AB

December 2010

Svensk Kärnbränslehantering AB
Swedish Nuclear Fuel
and Waste Management Co

Box 250, SE-101 24 Stockholm
Phone +46 8 459 84 00

ISSN 1402-3091
SKB R-11-01

Pandora – a simulation tool for safety assessments

Technical description and user's guide

Per-Anders Ekström, Facilia AB

December 2010

Keywords: SKBdoc 1269462.

A pdf version of this document can be downloaded from www.skb.se 2011-10.

Abstract

This report documents a flexible simulation tool, Pandora, used in several post closure safety assessments in both Sweden and Finland to assess the radiological dose to man due to releases from radioactive waste repositories. Pandora allows the user to build compartment models to represent the migration and fate of radionuclides in the environment. The tool simplifies the implementation and simulation of radioecological biosphere models in which there exist a large set of radionuclides and input variables.

Based on the well-known technical computing software MATLAB and especially its interactive graphical environment Simulink, Pandora receives many benefits. MATLAB/Simulink is a highly flexible tool used for simulations of practically any type of dynamic system; it is widely used, continuously maintained, and often upgraded. By basing the tool on this commercial software package, we gain both the graphical interface provided by Simulink, as well as the ability to access the advanced numerical equation solving routines in MATLAB. Since these numerical methods are well established and quality assured in their MATLAB implementation, the solution methods used in Pandora can be considered to have high level of quality assurance. The structure of Pandora provides clarity in the model format, which means the model itself assists its own documentation, since the model can be understood by inspecting its structure.

With the introduction of the external tool Pandas (Pandora assessment tool), version handling and an integrated way of performing the entire calculation chain has been added. Instead of being dependent on other commercial statistical software as @Risk for performing probabilistic assessments, they can now be performed within the tool.

Sammanfattning

I den här rapporten beskrivs ett flexibelt simuleringsverktyg, Pandora, vilket har använts i ett flertal slutförvarsprojekt i både Sverige och Finland för att bedöma de radiologiska effekter som kan uppstå vid eventuellt framtida läckage från förvar av radioaktivt avfall. Pandora ger användaren möjlighet att bygga box-modeller för att representera transport och ackumulation av radionuklider i miljön. Verktyget är speciellt framtaget för att förenkla skapande och simulering av radioekologiska biosfärsmodeller för ett stort antal radionuklider och styrvariabler.

Verktyget är baserat på den välkända tekniska beräkningsprogramvaran MATLAB och särskilt dess grafiska simuleringsverktyg Simulink. MATLAB/Simulink är ett välanvänt, mångfacetterat och ytterst flexibelt verktyg som kan användas för simuleringar av praktiskt taget alla typer av dynamiska system. Genom att basera verktyget på detta mjukvarupaket får Pandora tillgång till både det avancerade grafiska gränssnittet som tillhandahålls av Simulink och alla de numeriska metoder för att lösa avancerade ekvationer som tillhandahålls av MATLAB. Eftersom dessa numeriska metoder är väletablerade och kvalitetssäkrade i MATLAB kan också Pandora anses ha hög nivå av kvalitetssäkring. Strukturen på Pandora skapar klarhet i modellens format, vilket innebär att själva modellen står för sin egen dokumentation, eftersom modellen kan förstås genom att inspektera dess struktur.

Med införandet av det externa verktyget Pandas har Pandora tillgång till versionshantering och ett integrerat sätt att utföra hela beräkningskedjan. Istället för att vara beroende av kommersiell statistisk-programvara som @Risk för att utföra probabilistiska bedömningar och känslighetsanalyser kan de nu göras direkt i verktyget.

Contents

1 Introduction

The radioactive waste from nuclear power plants in Sweden is managed by the Swedish Nuclear Fuel and Waste Management Co, SKB. Spent nuclear fuel from the power plants is planned to be placed in a geological repository according to the KBS-3 method. In this method, copper canisters with a cast iron insert containing spent fuel are surrounded by bentonite clay and deposited at approximately 500 m depth in saturated granitic rock. Around 12,000 tonnes of spent nuclear fuel is forecasted to arise from the Swedish nuclear power programme, corresponding to roughly 6,000 canisters in a KBS-3 repository.

Between 2002 and 2007, SKB performed site investigations at two different sites along the eastern coast of southern Sweden; Forsmark in the municipality of Östhammar and Laxemar-Simpevarp in the municipality of Oskarshamn, with the intention of identifying a safe location for the repository. Data from the site investigations were used to produce a comprehensive, multi-disciplinary site description for each of the sites. The resulting site descriptions were reported in /SKB 2008/ (Forsmark) and /SKB 2009/ (Laxemar-Simpevarp). Based on available knowledge from the site descriptions and from preliminary safety assessments of the planned repository, SKB decided in June 2009 to submit an application to construct a repository in Forsmark. The location of Forsmark is shown in Figure 1-1. In 2011, an application for the construction of a geological repository for spent nuclear fuel at Forsmark is planned to be submitted.

According to the regulations from the Swedish Radiation Safety Authority, SSM, a safety assessment of the planned repository, evaluating features, events and processes that potentially may lead to the release of radionuclides, has to be performed before the construction of the repository is started (SSMFS 2008:21). The evaluation of the long-term safety of the repository is reported in the SR-Site main report /SKB 2011/ (SR-Site is the project of which the report is a product). Accordingly, it is an important supporting document to the application.

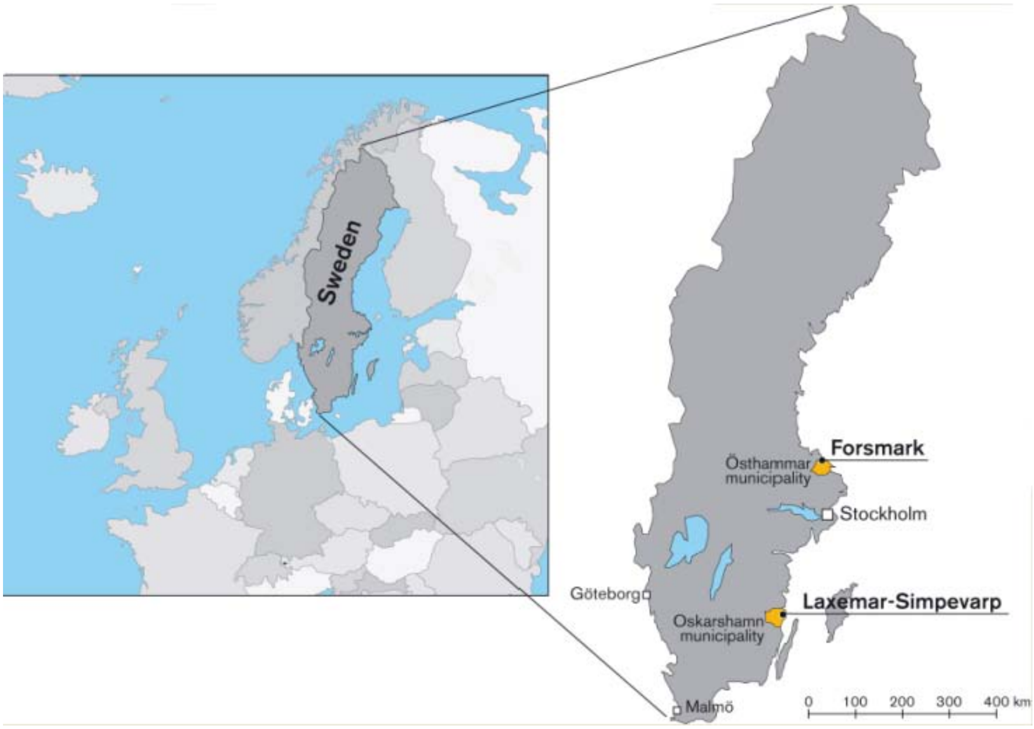


Figure 1-1. *Location of the Forsmark site.*

The safety assessment SR-Site, which is described in the SR-Site main report /SKB 2011/, is focused on three major fields of investigation: the performance of the repository, the geosphere and the biosphere. The biosphere part of SR-Site, SR-Site Biosphere, provides estimates for human exposure given a unit release (*Landscape Dose Conversion Factors*). Multiplying these factors with modelled release rates from the geosphere results in estimates of the annual doses used to assess compliance with the regulatory risk criterion. The effects on the environment of a potential release from the repository are also assessed in SR-Site Biosphere. This report is a product within SR-Site Biosphere and covers a description and user's guide of the simulation tool Pandora. The following sections describe the historical background of Pandora and give an overview of its predecessors.

Together with its predecessors, Pandora has been successfully applied within several safety assessments in both Sweden and Finland, the SKB preliminary biosphere safety case SR-Can bio for high level waste, see /Avila and Bergström 2006/ and /Avila 2006/. Pandora were also used within the SKB safety assessment for low and intermediate level waste SAR-08, see /Avila and Pröhl 2008/ and /Bergström et al. 2008/. In Finland Pandora were the main computational tool for the biosphere computations within the Posiva KBS-3H safety evaluation /Avila and Bergström 2007, Ekström and Broed 2006, Broed 2007, Broed 2008, Broed et al. 2009/.

Concepts in Pandora have been developed from an earlier tool, Tensit /Jones et al. 2004/. The implementation of Pandora where based on many fruitful ideas found developing and working with the Tensit tool. In addition, during the development years of Pandora the tool has grown with several new features. These features improve the user friendliness of the tool; add the possibility to solve models that are more complicated and to support the verification, quality control and version handling strategy.

The work done within the biosphere assessment in SR-Site has been conducted by a number of people. This group has been involved from the beginning of the site investigation, via the site characterization and modelling tasks, and in this final synthesis for the safety assessment SR-Site. The project members in alphabetical order, their affiliation and role are listed below:

Eva Andersson**, Studsvik Nuclear AB	process descriptions and limnic ecosystems
Karin Aquilonius, Studsvik Nuclear AB	marine ecosystems
Rodolfo Avila, Facilia AB	radionuclide modelling and dose assessment
Sten Berglund, SKB	hydrology and near-surface radionuclide transport
Emma Bosson, SKB	surface hydrology and near-surface hydrology
Lars Brydsten, Umeå University	regolith dynamics and lake development and GIS analysis
Anders Clarhäll, SKB	editorial work
Per-Anders Ekström, Facilia AB	numerical modelling
Anders Engqvist, KTH	oceanography
Sara Grolander, Facilia AB	site specific chemical data
Anna Hedenström, SGU	regolith
Ulrik Kautsky*, SKB	overall biosphere assessments project leader, scientific and method development
Tobias Lindborg*, SKB	site modelling and landscape development
Angelica Lorentzon af Ekenstam, SKB	process description and project administration
Anders Löfgren, EcoAnalytica	terrestrial ecosystems and synthesis
Sara Nordén, SKB	radionuclide and element specific properties
Peter Saetre, SKB	data evaluation, synthesis and review of results, editor of this report
Mona Sassner, DHI	surface hydrology
Gustav Sohlenius, SGU	regolith and future land use
Mårten Strömgren, Umeå University	GIS analysis, landscape development
Björn Söderbäck, SKB	site descriptions and historical development
Jesper Torudd, Facilia AB	non-human biota assessment
Mats Tröjbom, MTK AB	biogeochemistry and mass balances
Per-Gustav Åstrand, Facilia AB	numerical modelling

* project leader SR-Site Biosphere

** assistant project leader SR-Site Biosphere

2 This report

The Pandora software report (this report) is the document describing the simulation tool, Pandora, used to implement and simulate the transport and accumulation of potentially released radionuclides in biosphere objects for the biosphere analysis within the SR-Site Safety assessment /SKB 2011/. The development of Pandora was a joint collaboration between SKB and Posiva Oy /Jones et al. 2004/ to solve and document complicated biosphere models developed within their respective safety assessment project.

Biosphere models exist as a final component in the chain of models used in the safety assessment. The safety assessment describes how radionuclides are transferred through the Near Field (waste itself, canister, backfill and sealing materials) into the Far Field (geosphere) and then to the biosphere, where the radionuclides may result in dose to humans. Results from biosphere modelling are the estimated dose to individuals of the local population.

The purpose of this document is mainly to document the tool and work as a user's guide for the novice user. In addition, it reports some reimplementation of verification simulations as well as it provides some illustration of the code.

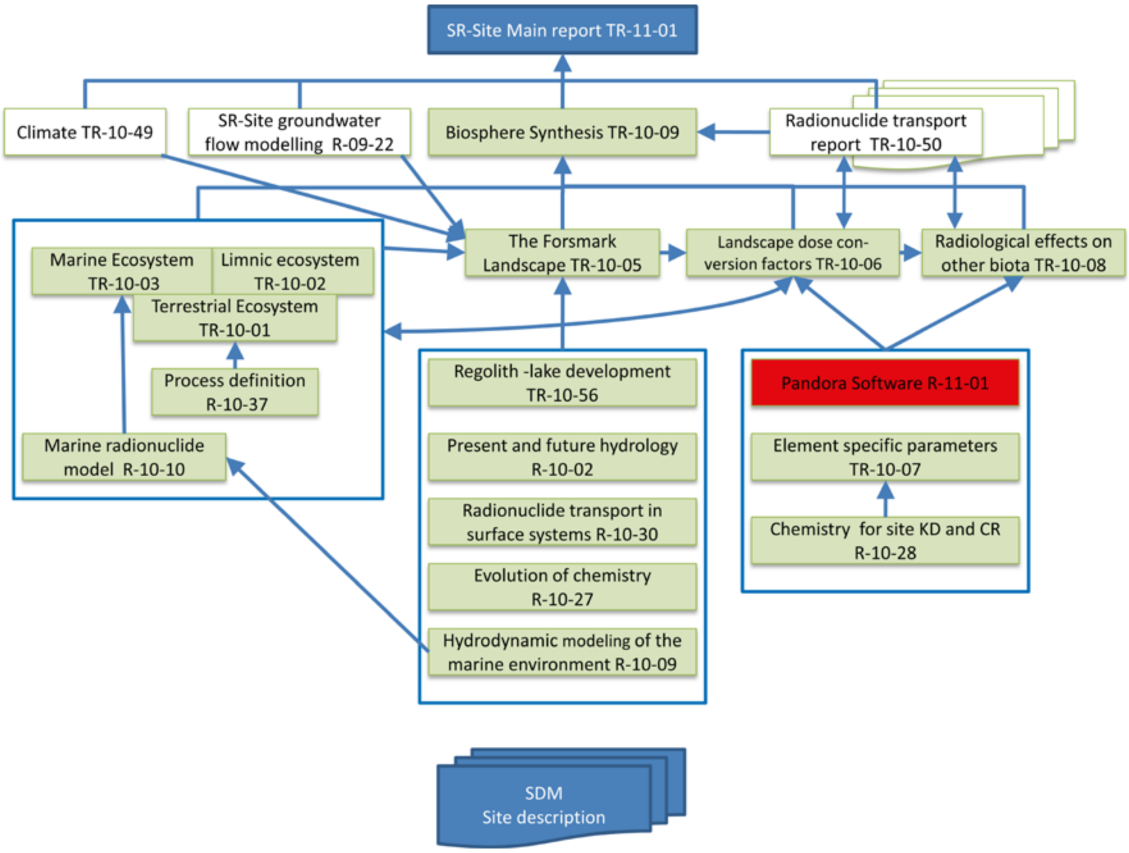

SR-Site Main report TR-11-01
Climate TR-10-49
SR-Site groundwater flow modelling R-09-22
Biosphere Synthesis TR-10-09
Radionuclide transport report TR-10-50
Marine Ecosystem TR-10-03
Limnic ecosystem TR-10-02
Terrestrial Ecosystem TR-10-01
Process definition R-10-37
Marine radionuclide model R-10-10
The Forsmark Landscape TR-10-05
Landscape dose conversion factors TR-10-06
Radiological effects on other biota TR-10-08
Regolith -lake development TR-10-56
Present and future hydrology R-10-02
Radionuclide transport in surface systems R-10-30
Evolution of chemistry R-10-27
Hydrodynamic modeling of the marine environment R-10-09
Pandora Software R-11-01
Element specific parameters TR-10-07
Chemistry for site KD and CR R-10-28
SDM
Site description

Figure 2-1. *The hierarchy of reports produced within the SR-Site Biosphere project. This report (market red) and its dependencies to biosphere reports (green) and other reports within SR-Site. Frames indicates groups of equal level reports. Arrows indicates major dependencies, but most reports interact with each other and the production of the information has been an iterative process. See text for a description of the contents of each report. The sources of data should be searched for in the sub reports if not explicitly pointed to in text. SDM is the site descriptive model (/Lindborg ed. 2010/ for the Forsmark site).*

The report is divided into chapters and sections that will guide the reader through the features of the Pandora tool and Pandas:

Chapter 1 gives a short *background* to the SR-Site safety assessment, a list of the persons involved in SR-Site Biosphere project and a background to the development of the software.

Chapter 2 (this chapter) puts this report in context within SR-Site.

Chapter 3 describes the Simulink block library (a set of utility blocks) that comprises the essence of the Pandora tool.

Chapter 4 goes into depth with the independent MATLAB user interface, called the Pandora Manager, which is a utility to facilitate the implementation of radionuclide transport models in Simulink using the Pandora block library.

Chapter 5 introduces the Java-based assessment tool, Pandas.

Chapter 6 has two systematic guides on how to create a Pandora model, tutorial style.

Chapter 7 presents results from verification simulations together with illustrations of implemented models in Pandora.

3 Pandora Block Library

Pandora has a set of elementary building blocks defined as a library within Simulink; this set of blocks is the core of Pandora. Simulink, developed by MathWorks (www.mathworks.com), is an environment for modelling, simulating and analysing dynamic systems. Its primary interface is an interactive graphical block-diagramming tool and has customizable block libraries, e.g. the Pandora block library. It offers tight integration with the rest of the MATLAB environment, providing immediate access to an extensive range of tools to develop algorithms, analyse and visualise simulations, create batch processing scripts, customise the modelling environment, and define signal, parameter, and test data. A Pandora model is a Simulink model with building blocks taken from the Pandora library. Typically, within a Pandora model, only a subset of all blocks in the Pandora library is used. In the subsequent sections the most commonly used blocks are described. In the last section, we also briefly describe the other blocks.

3.1 Pandora block

The Pandora block, see Figure 3-1, allows centralised selection and handling of multiple radionuclides to be included in a Pandora model. This block is a required component in any Pandora model since it defines the materials modelled. Arrange the modelled radionuclides to be included in the model in any desired order within this block. Signals in the model will then follow this order in all model blocks. In addition to the names of simulated radionuclides (or elements or species), the Pandora block also specifies half-lives (a vector) and decay chains (a matrix). For stable, non-decaying, materials, the usage of the value Inf (infinite) as half-life is necessary. For non-branching decay, a zero matrix should be used as the decay chain matrix.

Radionuclide properties can (preferably) be edited through the graphical interface called Pandora Manager, see Chapter 4. In any Pandora model, it is required to include one and only one Pandora block. When doing this, decay and in-growth data, as well as names of selected radionuclides become available throughout the entire model. The names of the radionuclides are not transferred in the model, but are instead read from the Radionuclide Manager. Therefore, there is no need or possibility to connect the Pandora block with other blocks within the model. In addition, the user can define the modelled unit to be either mole or Becquerel.

In the decay chain matrix, each column represents a parent to a daughter nuclide. Figure 3-2 illustrates how to set up the decay chain matrix having the defined ratios between parent and daughter nuclides.

Figure 3-3 shows the editable Mask dialog of the Pandora block, which is the dialog that opens up when double-clicking the block.

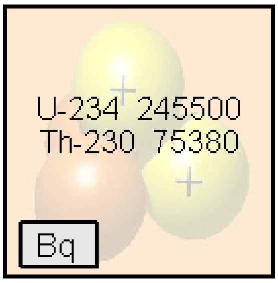


Figure 3-1. Illustration of the Pandora block.

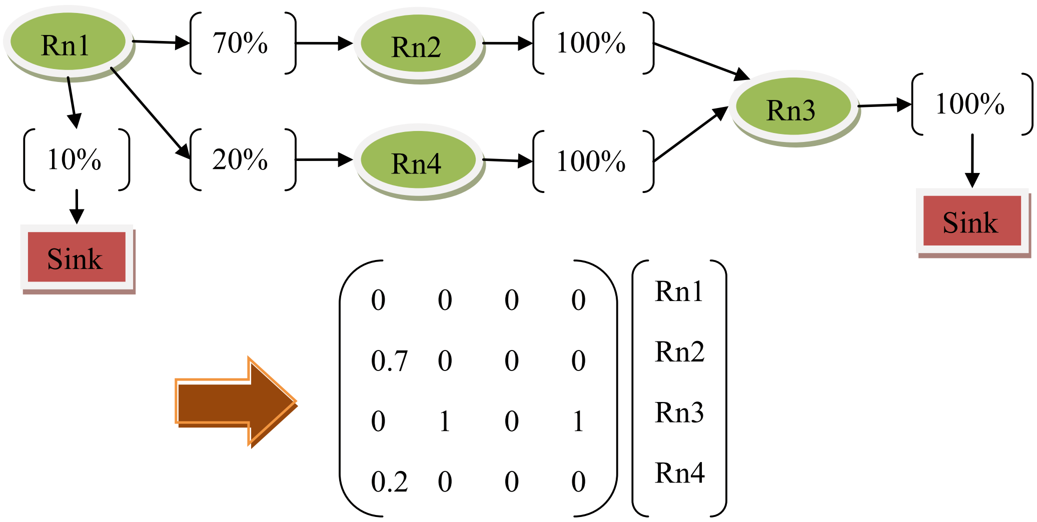
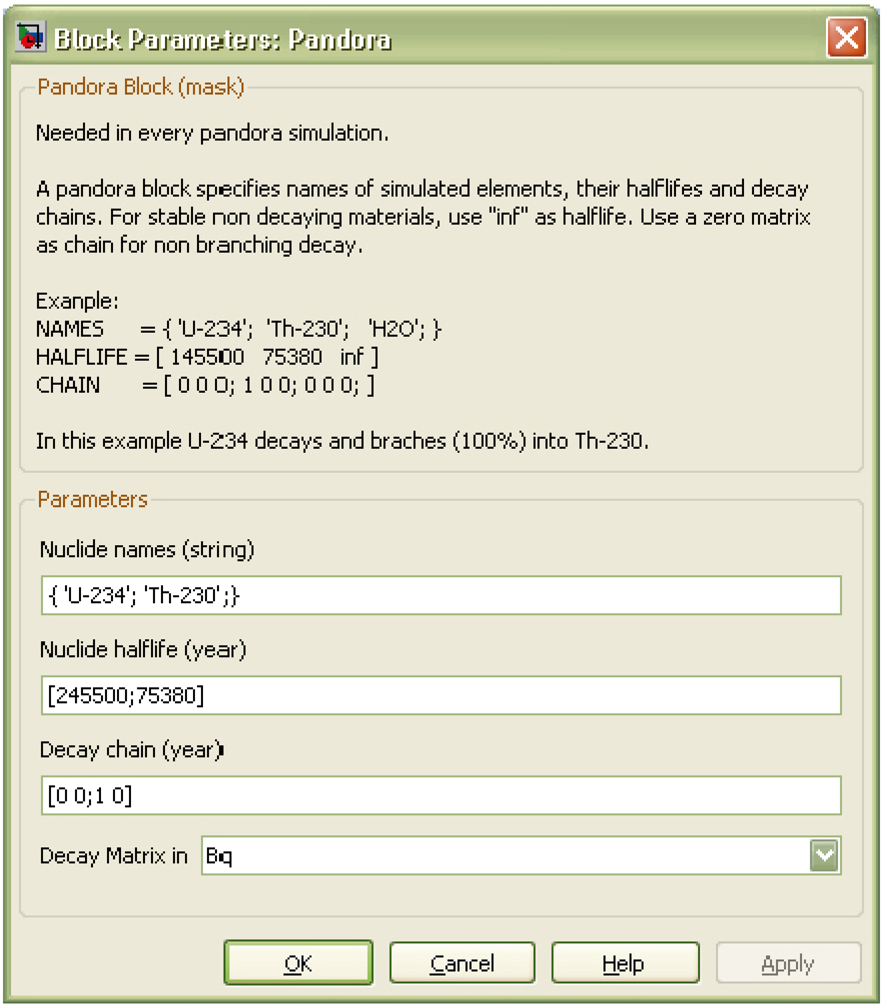


Figure 3-2. *Illustration of the creation of decay chain matrix based on decay ratios.*



Pandora Block (mask)

Needed in every pandora simulation.

A pandora block specifies names of simulated elements, their halflifes and decay chains. For stable non decaying materials, use "inf" as halflife. Use a zero matrix as chain for non branching decay.

Exanple:
NAMES = { 'U-234'; 'Th-230'; 'H2O'; }
HALFLIFE = [145500 75380 inf]
CHAIN = [0 0 0; 1 0 0; 0 0 0;]

In this example U-234 decays and braches (100%) into Th-230.

Parameters

Nuclide names (string)

{ 'U-234'; 'Th-230';}

Nuclide halflife (year)

[245500;75380]

Decay chain (year)

[0 0;1 0]

Decay Matrix in Bq

Figure 3-3. *Mask dialog of the Pandora block.*

3.2 Reservoir and transfer blocks

A Reservoir block represents a compartment in the system and is an integrator with support for removal of inventory through decay and leaching. This block together with the Pandora block will automatically model the decay and ingrowth of radionuclides. As shown in Figure 3-4, the Reservoir block depicted as a bucket, with an input shown on the left of the block and a "level" output on the right.

To model the transfer of material out from or between Reservoir blocks, the Transfer block is used. The Transfer block has two inputs: a "rate" input (which shows at the top of the input side of the Transfer block) and an unnamed lower input. To achieve an outgoing flux from the Reservoir block, connect the "level" output of the Reservoir block to the unnamed lower input of the Transfer block. The leaching rate from the Reservoir block is then defined by the signal connected to the upper in-port of the Transfer block. By default, the outgoing flux of materials from the Reservoir block at any time is the amount of inventory in the Reservoir block multiplied by the defined rate. If one does not want the transfer to depend on the Reservoir inventory, or one wants define this explicitly, uncheck the Use as rate-checkbox in the mask dialog of the Transfer block.

The Reservoir block has by default a filled bucket image representing the storage of inventory and the Transfer block is an arrow representing the leaching of inventory away from the Reservoir.

The Reservoir block in Pandora is more generic than in Tensit /Jones et al. 2004/. It can now accommodate an unlimited number of outgoing Transfers from the Reservoir; the number of outgoing Transfers will be shown on top of the Reservoir bucket icon. By default the colour of the transfer arrow is red, but once it is correctly connected to a Reservoir block and the model is updated it transforms into a gray colour.

Figure 3-5 illustrates how connections can be made between Reservoir blocks in a Pandora model using the Transfer block. Inside the Pandora block, the decay chain is defined such that U-234 decays and branches (100%) into Th-230. Blocks defined as yellow clovers are parameter blocks that depends on the elements in the Pandora block; they are called Element-Specific Parameter blocks and are described in Section 21. The rightmost block is called a Plot block; it stores the inventory in Reservoir B

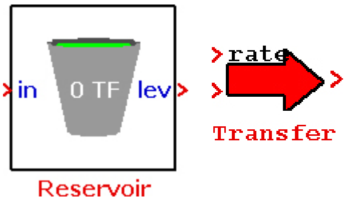


Figure 3-4. *Illustration of the Reservoir (bucket) to the left and the Transfer block (arrow) to the right.*

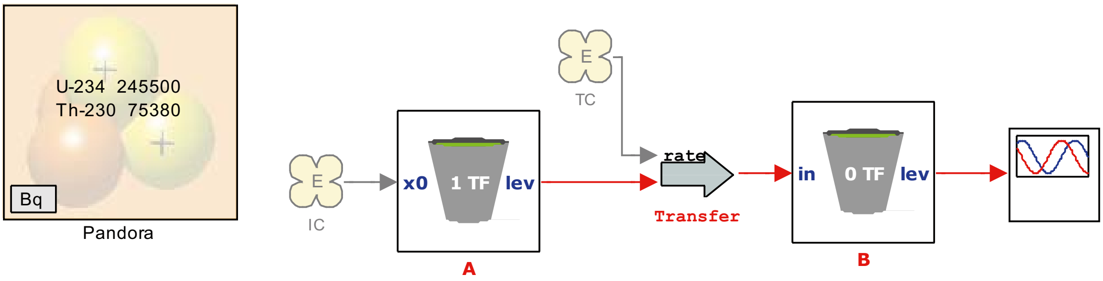


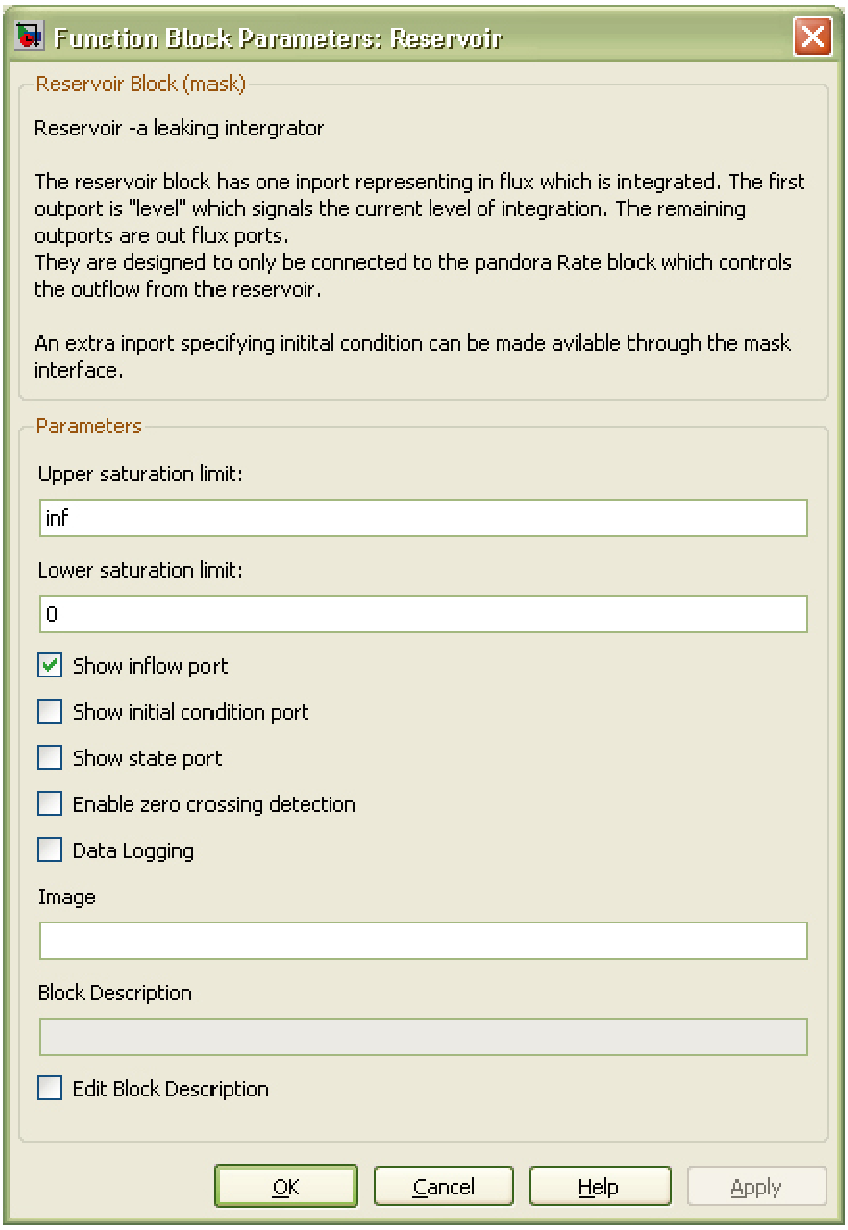
Figure 3-5. *Illustration of a simple two-compartment model.*

and allows creation of a customised MATLAB plot of the results. For more information about the Plot block, refer to the Pandora Manual. This simple model corresponds to the following system of differential equations:

$$\frac{dA_U^{234}}{dt} = -TC_U^{234}A_U^{234} - \frac{\ln 2}{245500}A_U^{234}$$
$$\frac{dA_{Th}^{230}}{dt} = -TC_{Th}^{230}A_{Th}^{230} - \frac{\ln 2}{75380}A_{Th}^{230} + \frac{\ln 2}{245500}A_U^{234}$$
$$\frac{dB_U^{234}}{dt} = -TC_U^{234}A_U^{234} - \frac{\ln 2}{245500}B_U^{234}$$
$$\frac{dB_{Th}^{230}}{dt} = -TC_{Th}^{230}A_{Th}^{230} - \frac{\ln 2}{75380}B_{Th}^{230} + \frac{\ln 2}{245500}B_U^{234}$$

using initial conditions for the Reservoir block A defined with the element specific parameter block called IC, by default the initial condition in Reservoir block B is zero.

Figure 3-6 and Figure 3-7 illustrates the Mask dialogs of the Reservoir and Transfer blocks respectively.



Reservoir Block (mask)

Reservoir -a leaking intergrator

The reservoir block has one inport representing in flux which is integrated. The first outport is "level" which signals the current level of integration. The remaining outports are out flux ports.
They are designed to only be connected to the pandora Rate block which controls the outflow from the reservoir.

An extra inport specifying initital condition can be made avilable through the mask interface.

Parameters

Upper saturation limit: inf

Lower saturation limit: 0

☑ Show inflow port

☐ Show initial condition port

☐ Show state port

☐ Enable zero crossing detection

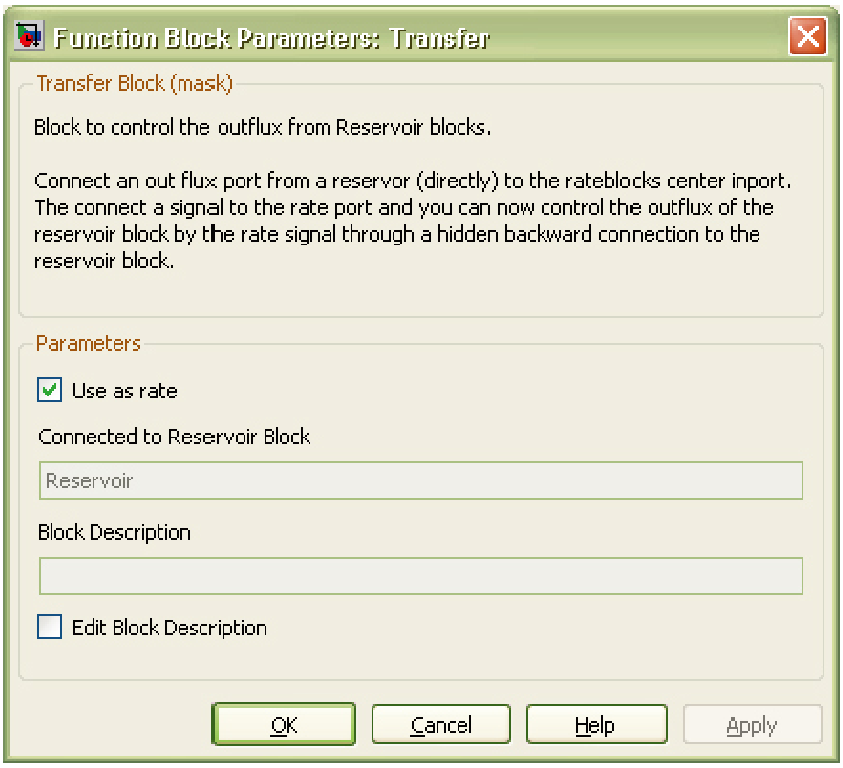
☐ Data Logging

Image:

Block Description:

☐ Edit Block Description

Figure 3-6. *Mask dialog of the Reservoir block.*



Transfer Block (mask)

Block to control the outflux from Reservoir blocks.

Connect an out flux port from a reservor (directly) to the rateblocks center inport. The connect a signal to the rate port and you can now control the outflux of the reservoir block by the rate signal through a hidden backward connection to the reservoir block.

Parameters

☑ Use as rate

Connected to Reservoir Block

Reservoir

Block Description

☐ Edit Block Description

Figure 3-7. *Mask dialog of the Transfer block.*

3.3 Function block

In a Pandora model, one usually needs to solve multiple equations with several input variables. In Simulink there are two standard ways to solve equations: either use the built-in “general expression block”, or build the equation by hand using elementary building blocks.

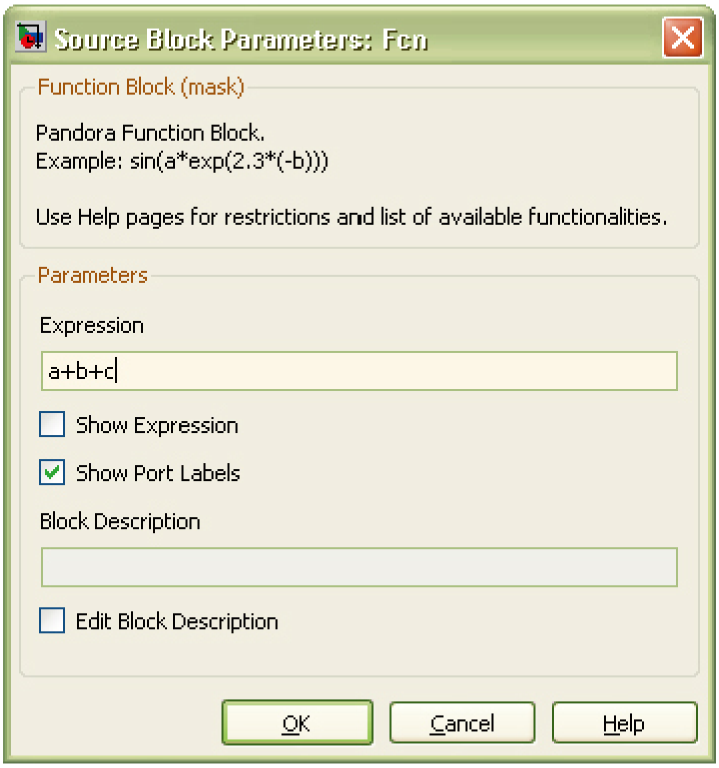
The “general expression block” takes only one input, a vectorised signal (u), so the equation has to be written using indexing of this one input variable (u(1), u(2),…) instead of using the real variable names. Another more serious problem is that due to the hard-coded indexing to obtain input variables for the equation, the block cannot automatically handle signal-dimension changes when modifying the number of modelled species in the Pandora block.

Building each equation manually by hand using elementary building blocks can be tedious, and such tedious work can be prone to input errors. In addition, there is the potential for the model to be cluttered with elementary blocks and connections that rather should be hidden.

The Function block is a vectorised version of its built-in counterpart the “General Expression block”. The Function block parses a MATLAB style equation and builds up the solution by elementary blocks hidden underneath its subsystem. All unknown variables found in the equation will be added as inputs to the block. The Function block is therefore an automatic translator of an expression into individual Simulink blocks. The Function block has four main advantages over other possible approaches to solving equations: speed, readability, quality assurance and vectorised signals.
The simple interface of the Function block is illustrated by the Mask dialog in Figure 3-8.

In the expanded part of Figure 3-9 are a series of icons showing a Function block as it appears to the user. This Function block represents the equation **sin(a*exp(2.3*(–b)))**. The blocks and connections to the right illustrate how the block has internally added and connected building blocks to solve this simple equation.

To reduce conflicts, unary minuses are only allowed either in the beginning of expressions, as the first element of the expression, or together with a left parenthesis (i.e. **a+(–b)** is correct. **a+–b** is not correct). In the following tables, there are listings of the functions and operators supported in the Function block.



Source Block Parameters: Fcn

Function Block (mask)

Pandora Function Block.
Example: sin(a*exp(2.3*(-b)))

Use Help pages for restrictions and list of available functionalities.

Parameters

Expression

a+b+c

□ Show Expression

☑ Show Port Labels

Block Description

□ Edit Block Description

Figure 3-8. *Mask dialog of the Function block.*

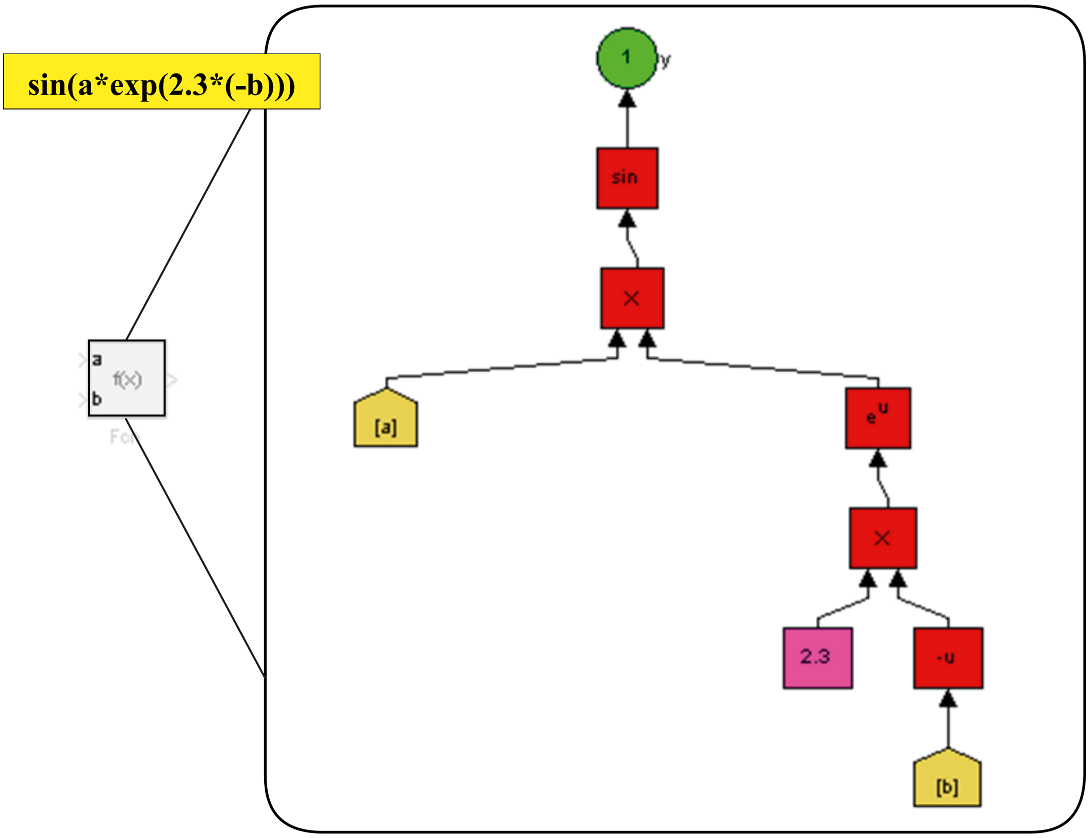


Figure 3-9. *Function block with simple equation and its underlying implementation.*

Table 3-1. Operator precedence – the Function block observes the following rules of operator precedence.

1	( )
2	^
3	* /
4	+ –
5	> < <= >=
6	== ~=

Table 3-2. Standard operators.

^	Power
*	Product
/	Division
–	Subtraction
+	Addition

Table 3-3. Relational operators.

<=	Less or equal to
>=	Larger or equal to
<	Less than
>	Larger than
==	Equal to
~=	Not equal to

Table 3-4. Trigonometric functions – trigonometric functions perform numerous common trigonometric functions. The atan2-function has two inputs. The first (upper) input is the y-axis or complex part of the function argument. The second (lower) input is the x-axis or real part of the function argument.

Trigonometric functions						
sin(x)	cos(x)	tan(x)	asin(x)	acos(x)	atan(x)	atan2(y,x)
Hyperbolic functions						
sinh(x)	cosh(x)	tanh(x)	asinh(x)	acosh(x)	atanh(x)	

Table 3-5. Logical operator functions – logical operators perform the specified logical operation on its inputs. An input value is TRUE (1) if it is nonzero and FALSE (0) if it is zero.

AND(x1,x2)	TRUE if all inputs x1 and x2 are TRUE
OR(x1,x2)	TRUE if at least one input of x1 and x2 is TRUE
NAND(x1,x2)	TRUE if at least one input of x1 and x2 is FALSE
NOR(x1,x2)	TRUE when no inputs x1 and x2 are TRUE
XOR(x1,x2)	if an odd number of inputs x1 and x2 are TRUE
NOT(x)	TRUE if the input x is FALSE

Table 3-6. Rounding functions – rounding functions apply a rounding function to the input signal to produce the output signal.

floor(x)	Rounds x to the nearest integer value towards minus infinity
ceil(x)	Rounds x to the nearest integer towards positive infinity
round(x)	Rounds x to the nearest integer
fix(x)	Rounds x to the nearest integer towards zero

Table 3-7. Continuous functions.

integrate(x1,x2)	Integrate input x1 using x2 as initial condition
derivate(x)	Derivate input x
delay(x1,x2)	Delay input x1 with x2 time-steps

Table 3-8. Mathematical functions – mathematical functions including logarithmic, exponential and modulus.

exp(x)	Exponent of the input x
log(x)	Natural logarithm of the input x
log10(x)	Base 10 logarithm of the input x
square(x)	Square of the input x
sqrt(x)	Square-root of the input x
rem(x1,x2)	Remainder of x1 after division by x2
mod(x1,x2)	Modulus of x1 after division by x2

Table 3-9. Miscellaneous functions.

sign(x)	Indicate the sign of the input x
abs(x)	Absolute value of the input x
width(x)	Width of the input signal x

3.4 Parameter and time-series blocks

A shortcoming of Simulink is the absence of a well-defined input parameter structure. Since Pandora models often tend to use large amount of input variables it is essential to have a standardized way to manage them.

In Pandora, there is a differentiation of input variables between being static and time evolving, using the concept names Parameter and Timeseries; a Parameter represents a constant value over the whole simulation period whilst the Timeseries can take on different values over time.

A further division of parameters as well as time-series by the following three categories are used:

- Site-Specific, illustrated by a circle with a capital S.
- Element-Specific, illustrated by a clover with a capital E.
- Universal Constant, illustrated by a cylinder with a capital U.

A Site-Specific variable block defines a scalar variable representative of the specified site. In the Pandora/Simulink context, this leads to the variable being unique inside a specific subsystem. As a result, a Site-specific block for the same physical variable inside two different subsystems will generate two different parameter variables. An Element-Specific block defines a variable that has values representative of each chemical element defined in the Pandora block. If the model contains more than one element, this leads to a vector of values. Within the model, element-Specific blocks are defined globally. This means that if two Element-Specific blocks are used with the same name in two different subsystems, they still will be the same variable, using the same parameter value. A Universal Constant is a scalar variable that is also globally defined, that is; it is always the same throughout the model. Figure 3-10 shows the icon representations of the different kinds of input variables found in Pandora. In Figure 3-11 the Mask dialog of the parameter block together with its different settings is illustrated.

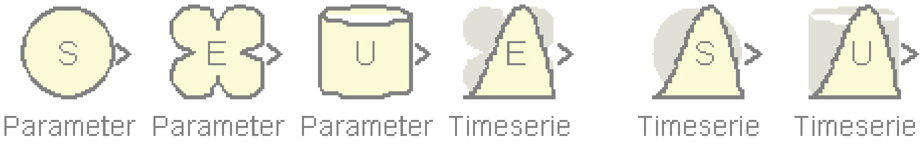
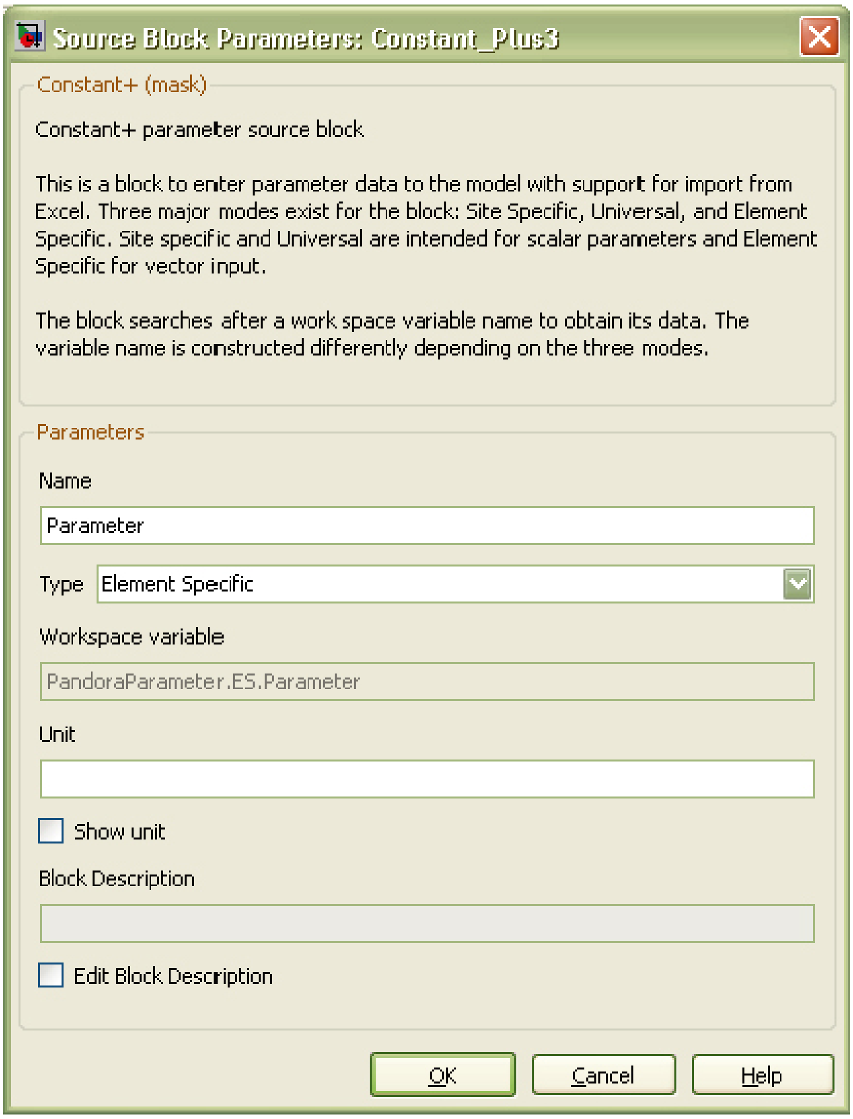


Figure 3-10. Illustration of the set of input variable blocks in Pandora. From left to right we have the Site-specific, Element-specific and Universal constant parameter blocks, and then we have the Element-specific, Site-specific and Universal constant time-series blocks.



Source Block Parameters: Constant_Plus3

Constant+ (mask)

Constant+ parameter source block

This is a block to enter parameter data to the model with support for import from Excel. Three major modes exist for the block: Site Specific, Universal, and Element Specific. Site specific and Universal are intended for scalar parameters and Element Specific for vector input.

The block searches after a work space variable name to obtain its data. The variable name is constructed differently depending on the three modes.

Parameters

Name: Parameter

Type: Element Specific

Workspace variable: PandoraParameter.ES.Parameter

Unit:

□ Show unit

Block Description:

□ Edit Block Description

Figure 3-11. *Mask dialog of a Parameter block.*

3.5 Other blocks

Apart from the set of most commonly used blocks listed in previous sections, Pandora has more usable blocks that might not exist in every model but can come in handy in different situations. In this section, we describe these blocks briefly.

3.5.1 Interface block

The Interface Block has no computational meaning, the signal connected to the input of the block will go unaffected out from the block. The reason to use this block is to name the signal coming from non-Pandora blocks so that, for example, the function block can manage the correct names on its in-ports. As with most other Pandora blocks, the Interface block also stores a block description. Figure 3-12 illustrates the diamond shape of the Interface block.

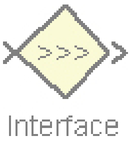


Figure 3-12. *Illustration of the Interface block.*

3.5.2 Filter and Selector blocks

The Filter block simplifies the implementation of models with different transfer functions for different nuclides, by setting the value of the unwanted nuclides to zero. The Selector block is similar to the Filter block but the unwanted nuclides are removed totally from the signal vector, thus the signal dimension is reduced in a selector. Note that Pandora does not keep track of the radionuclide order after a Selector block, this is up to the user.

The example model illustrated in Figure 3-13 portrays the use of Filter and Selector blocks: Display 1 shows the result of the use of a Selector block that reduces the dimensionality of the signal vector, now it holds only the selected signals (Th-230 and U-234). Result of the use of Filter block are illustrated with Display2 showing that the dimensionality of the signal vector is preserved but the signal values for the unselected nuclide (Cs-137) are set to zero.

3.5.3 Unit conversion blocks

The unit conversion blocks (bq2mole and mole2bq) are two utility blocks that provide conversion from activity in Becquerel to amount of substance in mole and vice versa. Unit conversion follows the two equations:

$$Mole = \frac{Bq \cdot HALFLIFE \cdot 60 \cdot 60 \cdot 24 \cdot 365}{\ln 2 \cdot N_A},$$

$$Bq = \frac{Mole \cdot \ln 2 \cdot N_A}{HALFLIFE \cdot 60 \cdot 60 \cdot 24 \cdot 365}$$

where HALFLIFE is taken from the Radionuclide Manager block (Pandora block) and is the half-life of each radionuclide. N_A is the Avogadro's number ($6.022 \cdot 10^{23}$). Figure 3-14 illustrates the simple look of the two conversion blocks.

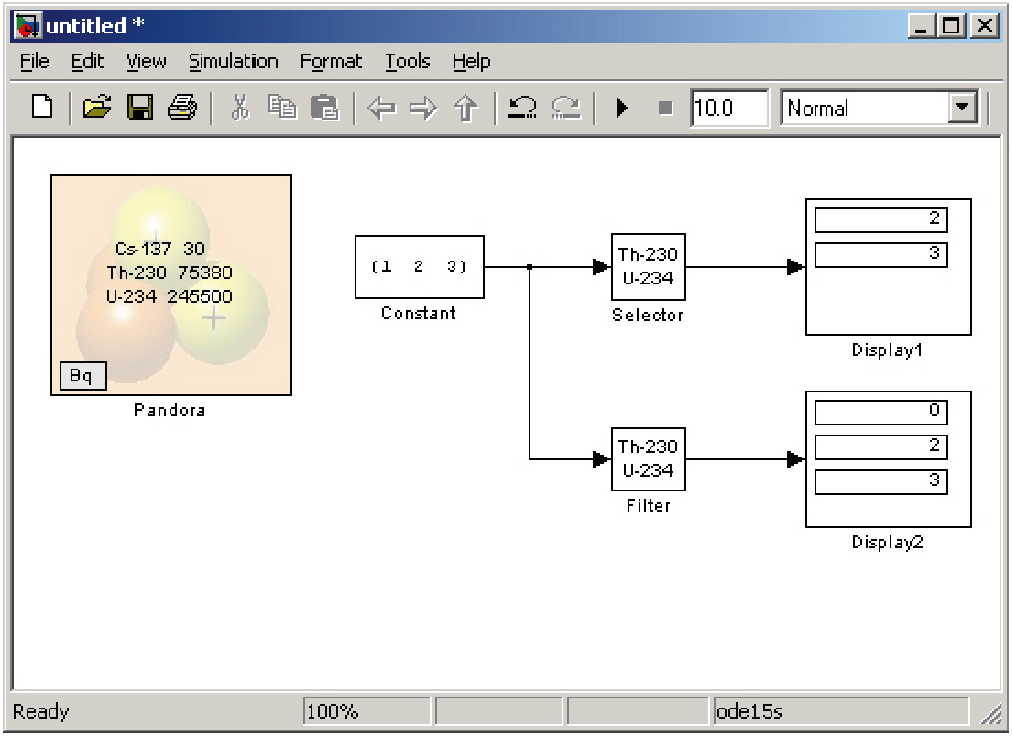


Figure 3-13. Example of the use of Filter and Selector blocks.

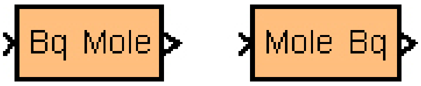


Figure 3-14. Illustration of Becquerel to Mole and Mole to Becquerel blocks.

3.5.4 Result and Plot blocks

Connect a line to the Result block to save the signal passing through this line to an output structure stored in the MATLAB workspace (Simulink logging signal). After a simulation, the signal can be analysed using the Results Viewer described in Section 4.8. The representation of the block is by the image of an old-fashioned 3½-inch floppy disk as can be seen to the right in Figure 3-15.

In Figure 3-15, we also see the Plot block represented by an image of a stylised oscilloscope. Connect a line to the Plot block to store the signal and export the data to a MATLAB graphics plot. After a simulation, double-click on block to obtain time plot. Right-click on block, choose Mask Parameters… to be able to change settings (mask parameters) of the block. See Figure 3-16 for settings of the Plot block.

Note that if input signal is nuclide dependent it is required to have the Nuclide Dependent check-box checked.

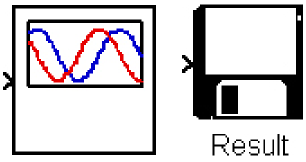
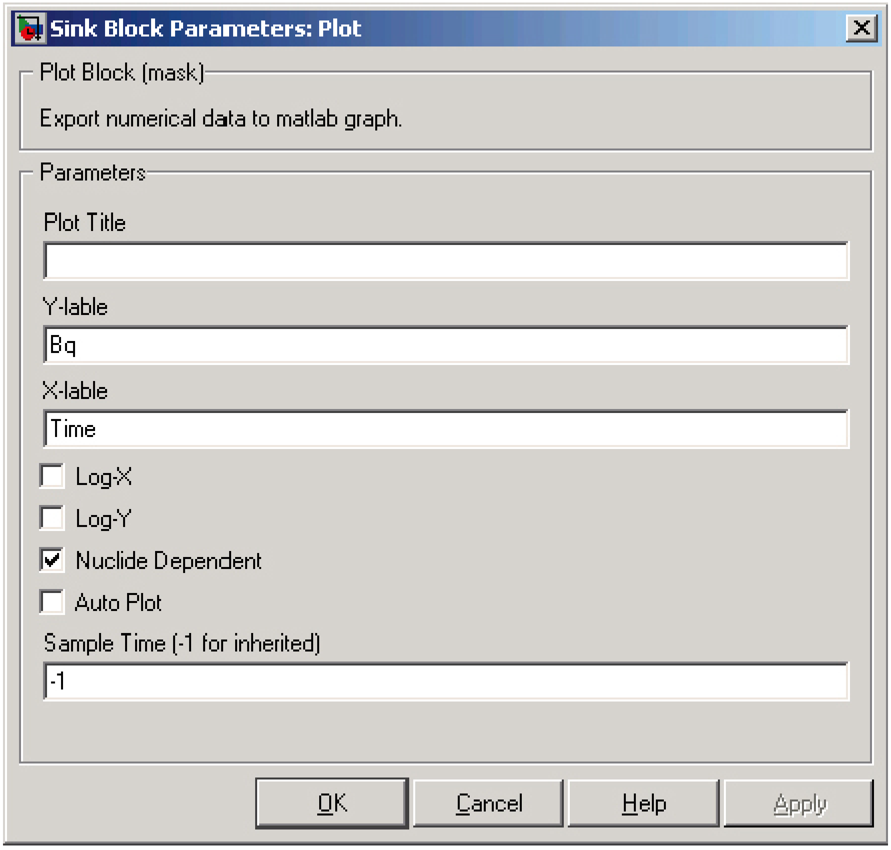


Figure 3-15. Illustration of Plot (oscilloscope) to the left and the Result block (floppy disk) to the right.



Sink Block Parameters: Plot

Plot Block (mask)

Export numerical data to matlab graph.

Parameters

Plot Title

Y-lable

Bq

X-lable

Time

□ Log-X

□ Log-Y

☑ Nuclide Dependent

□ Auto Plot

Sample Time (-1 for inherited)

-1

Figure 3-16. Mask dialog of Plot block.

3.5.5 Post-processing block

The Post-processing block is used to define variables to be computed after a simulation and analysed using the Result Viewer, see Section 4.8. Figure 3-17 shows the block and Figure 3-18 shows the interface of the settings dialog.

In the settings dialog one can add, edit and delete post-processing variables. A post-processing variable has a name and an expression (equation) and can have a comment. To help with defining the expression, possible variables that can be used within the expression are listed in the Variables text-box. A keypad is also available for a point-and-click style of inputting the equation.

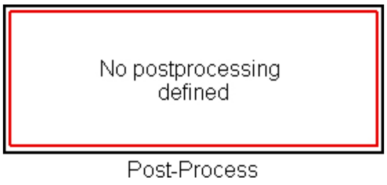
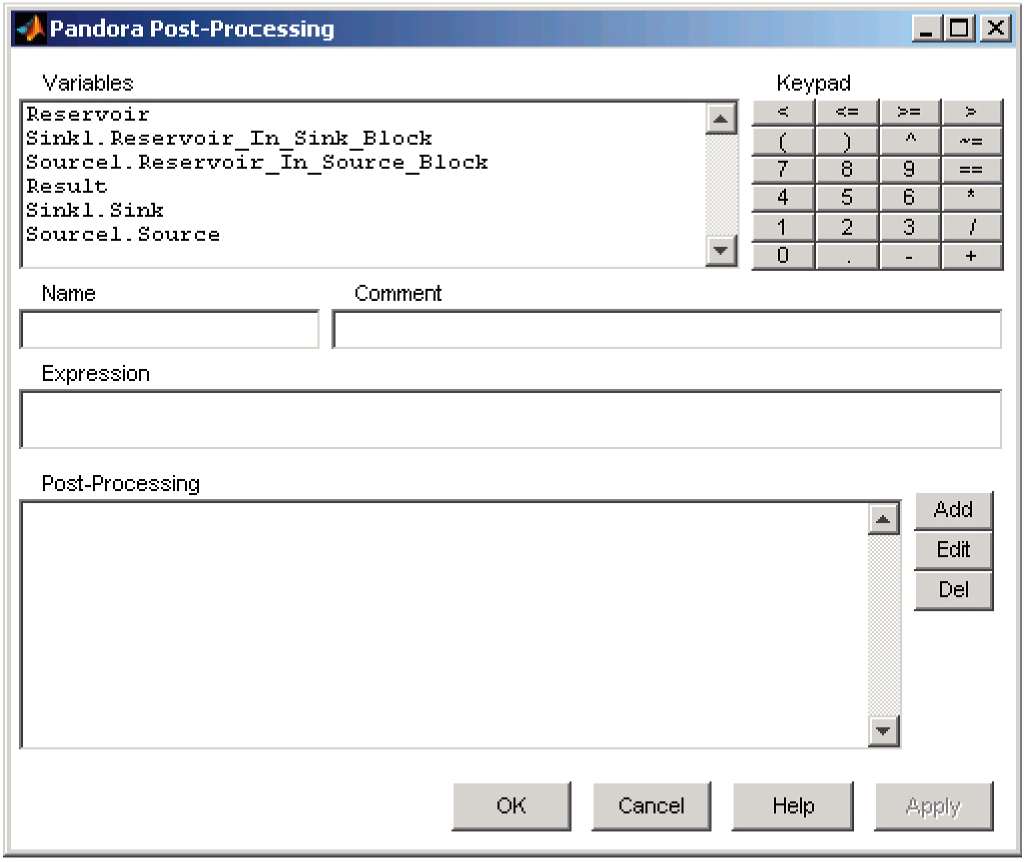


Figure 3-17. Illustration of Post-Processing block.



```
Reservoir
Sink1.Reservoir_In_Sink_Block
Source1.Reservoir_In_Source_Block
Result
Sink1.Sink
Source1.Source
```

Figure 3-18. Post-Processing settings dialog.

3.5.6 Subversion block

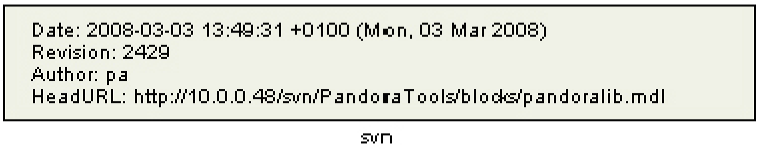
To secure traceability in projects, Pandora as well as Pandas, see Chapter 5, has support for a version handling system (SubVersion, /http://svnbook.red-bean.com/). In the SR-Site Biosphere project, SubVersion handles versions of raw data, parameters, models, and results. The Subversion block in Pandora shows information about the file-status in the subversion repository. Every time the model file is committed, the following four keywords are updated: Date, Revision, Author and HeadURL. The block is exchanging the specific keywords within the block at commit-stage. Figure 3-19 illustrates how the block looks like after a commit into a subversion repository. Presented are the exact date, revision, author and URL of the file of the version handled Pandora model.

For this block to work, one needs to add the specific properties as keywords to the file that is in subversion. Using the Tortoise SVN client (http://tortoisesvn.tigris.org/) one may add these keywords by right-click at the file under TortoiseSVN-field choose Properties.

At the interface, see Figure 3-20, choose 'Add…'.

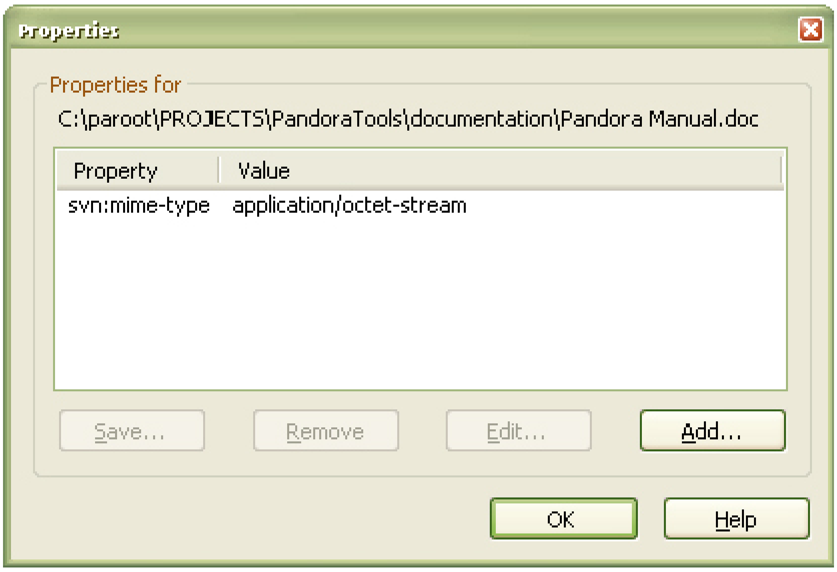
In the Property name drop-down menu, see Figure 3-21, choose svn:keywords and in the Property value list-box add Date, Revision, Author and HeadURL.

These settings can also be added recursively on a whole folder. Do the same as for a file but also click the Apply property recursively check-box. The Property name svn:keywords will not show in the drop-down menu, just add it manually using the keyboard.



Date: 2008-03-03 13:49:31 +0100 (Mon, 03 Mar 2008)
Revision: 2429
Author: pa
HeadURL: http://10.0.0.48/svn/PandoraTools/blocks/pandoralib.mdl

Figure 3-19. Illustration of Subversion block.



C:\paroot\PROJECTS\PandoraTools\documentation\Pandora Manual.doc

Property	Value
svn:mime-type	application/octet-stream

Figure 3-20. Subversion Properties interface.

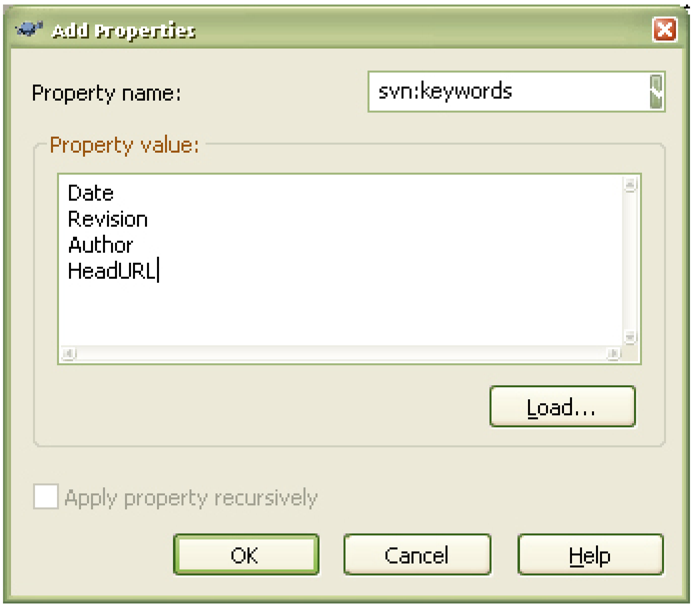


Figure 3-21. Subversion Add Properties interface.

3.5.7 Parameter storage block

Use the Parameter storage block to store parameter values in-between simulation sessions. Place this block in your model and when parameter values are in MATLAB workspace it is possible to store these values inside the block using the Pandora Manager described in Chapter 4. In the Parameter-menu in the Pandora Manager, choose Parameter Storage|FromWorkspace to import data from workspace to the block. Choose Parameter Storage|ToWorkspace to export the data inside the block to workspace. Figure 3-22 illustrates the upside down cone that represents the Parameter storage block.

The ToWorkspace feature can also be automated either by checking the box in the mask dialog or by doing the same from the Pandora Manager, see Chapter 4. If the check-box is checked it will say "on" on the block and at runtime the parameter data stored inside the block will automatically be copied out to workspace.

3.5.8 Source and Sink blocks

To keep track of the mass balance of the system, use the Source and Sink blocks (Figure 3-23) to store inflow and loss of material in model. To compute the mass balance of the system one needs to account for all inventory existing in the system, all inventory added to the system as well as all inventory that has left the system. To keep track of this one need to use Source blocks for all influxes to the system as well as all-eventual initial conditions and Sink blocks for all fluxes leaving the system. Data Logging needs to be checked in all Reservoir blocks as well as for all the Source and Sink blocks in the system. This is done in order to be able to store and sum the entire inventory that is currently in the system, that has been added to the system and that has left the system.

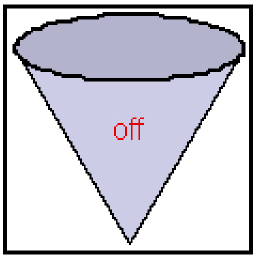


Figure 3-22. Illustration of Parameter storage block.

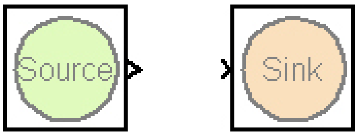


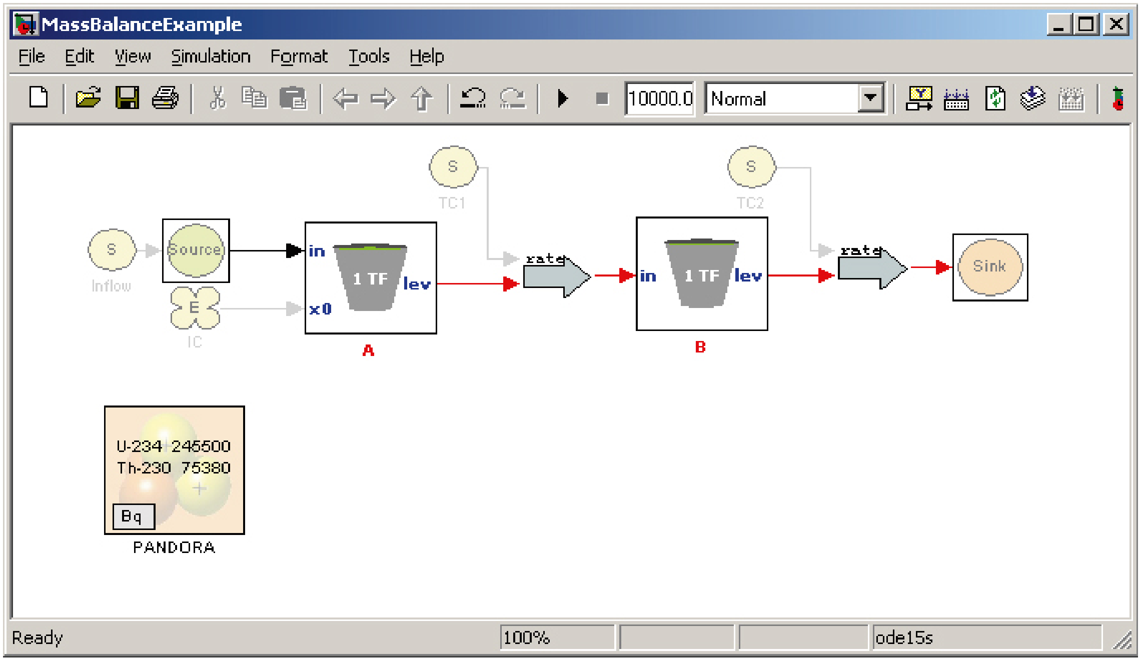
Figure 3-23. *Illustration of Source and Sink blocks.*

In the Source block, one can choose either to use external or internal input. Signal Logging in the Data Import/Export choice in the Configuration Parameters needs to be checked. After a simulation, use Tools|Mass Balance to get a chart over the absolute and relative mass balance over time. Absolute mass balance is the sum of all inventories and the sum of all sinks minus the sum of all sources. The relative mass balance is the absolute mass balance divided by the sum of the absolute value of everything divided by two.

Figure 3-24 illustrates an example model with the “Mass Balance blocks”. Here we have an initial condition and an inflow to reservoir A, a flux from A to B and a flux from B out of the system. Source blocks take care of the input and a Sink block takes care of the flow out from the system.

System is simulated over 10,000 years, with parameter values as follows: Inflow = 1e–5Bq/y, TC1 = 1e–3 1/y, TC2 = 1e–4 1/y, IC[Th-230] = 0 Bq and IC[U-234] = 1e2 Bq. With default simulation settings (relative tolerance of 1e-3 and absolute tolerance set to auto (1e-6)) together with the default Pandora block values we can generate the plots in Figure 3-25 and Figure 3-26 by choosing Tools|Mass Balance from the Pandora Manager after a simulation using the two different Simulink solvers ODE45 and ODE15S respectively.

Note: ODE45 has much better accuracy than ODE15S with this simple model using the same tolerance levels.



10000.0 Normal

U-234 245500
Th-230 75380
Bq

Ready 100% ode15s

Figure 3-24. *Mass Balance example model.*

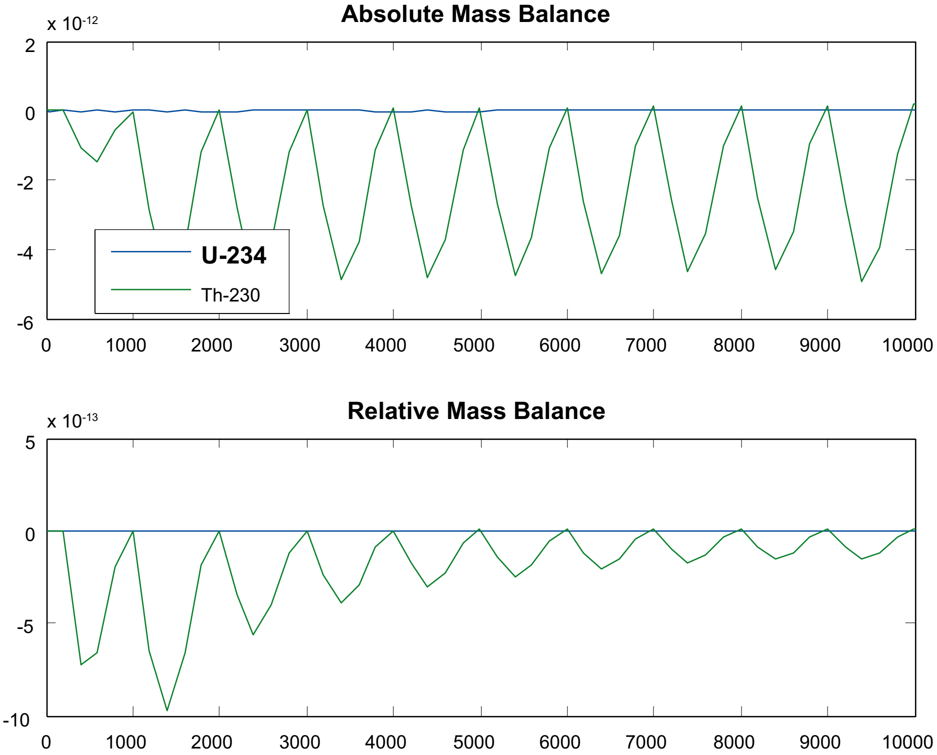


Figure 3-25. *Mass Balance of model MassBalanceExample using Simulink solver ODE45.*

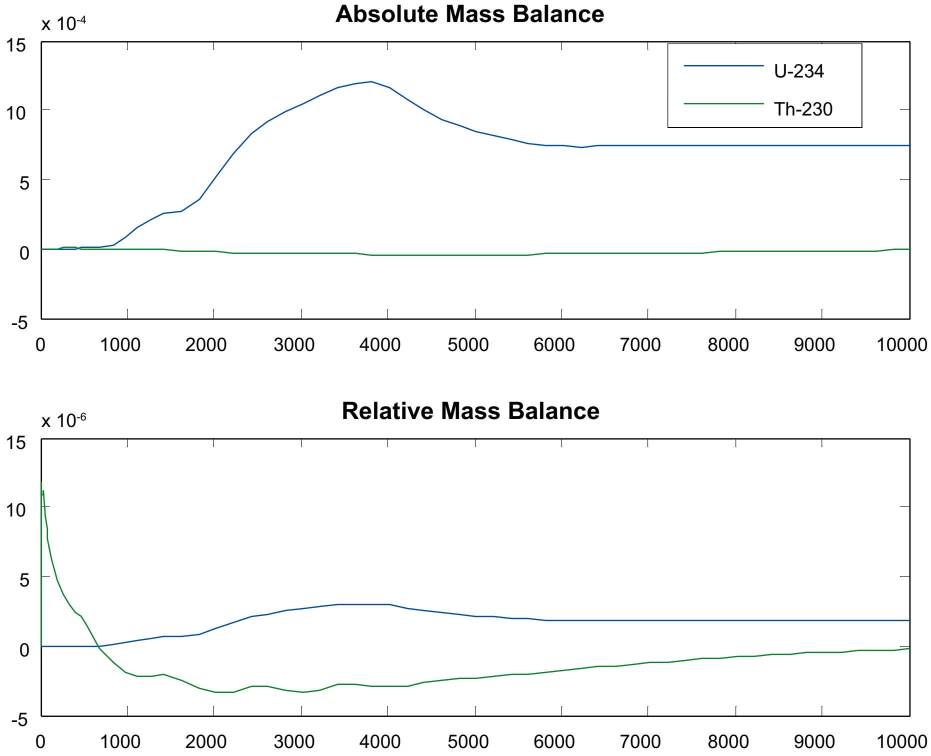


Figure 3-26. *Mass Balance of model MassBalanceExample using Simulink solver ODE15S.*

4 Pandora Manager

The Pandora Manager is a tool to aid model building and simulation. It is a MATLAB interface with four different perspectives: Nuclide perspective, Sub-system perspective, Function perspective and Parameter perspective. It has buttons for quickly creating a new Pandora model, opening Pandora models, opening the Simulink Library Browser, zooming Pandora models, Ghosting of Reservoir and Transfer blocks, Simulating Pandora models and opening the Results Viewer. The blocks in the Pandora block library can be used independently of the Pandora Manager, but the manager extensively facilitate the construction, simulation and maintenance of both smaller and larger models.

Not all features accessible from the Pandora Manager are described here. Described in the following sections are the key components.

4.1 Software perspectives

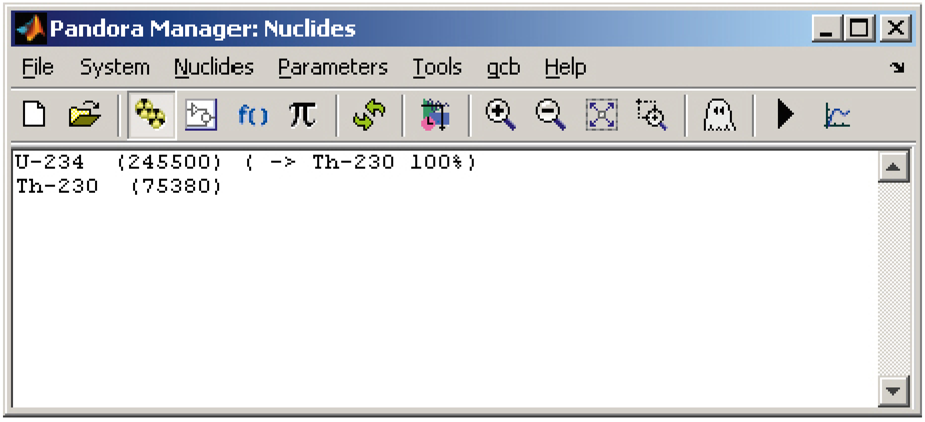
The intention of the four different perspectives (Nuclide, Sub-System, Function and Parameter) is to quickly get an overview over the current model from different aspects. Unfortunately, there is no automatic synchronisation between the Pandora Manager and the currently loaded Pandora model. For the Pandora Manager to synchronize with the current model, the refresh button (⟳) must be clicked to get updated perspectives. When refreshed; all four perspectives are updated according to the current state of the loaded model.

4.1.1 Nuclide

In the Nuclide perspective (☢), see Figure 4-1, an overview of the modelled materials is presented. A list of the nuclides included in the current Pandora model is shown, including half-lives and decay products together, with the branching percentage.

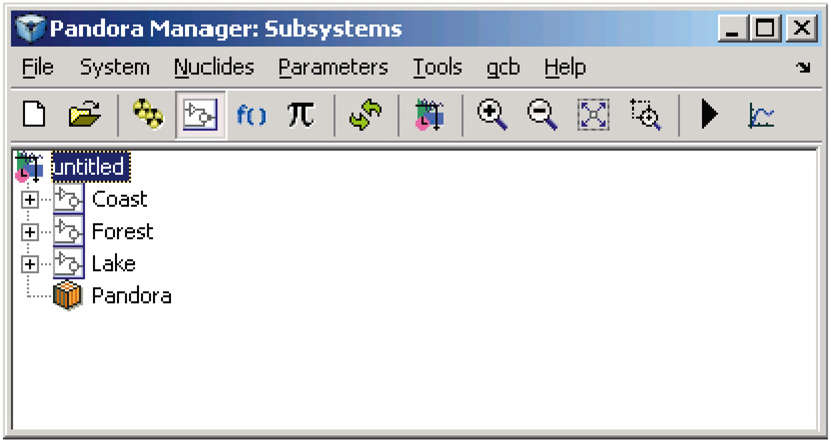
4.1.2 Sub-system

In the Sub-system perspective (▣), see Figure 4-2, the model hierarchy can be browsed as a folder tree. All Pandora blocks are shown in the tree and one can navigate to the block of interest in the model by right-clicking in the tree and choosing open. The sub-system of the chosen block will open and the block will be highlighted.



```
U-234   (245500)  ( -> Th-230 100%)
Th-230   (75380)
```

Figure 4-1. Pandora Manager with the Nuclide perspective selected.


Pandora Manager: Subsystems
File System Nuclides Parameters Tools gcb Help

untitled
Coast
Forest
Lake
Pandora

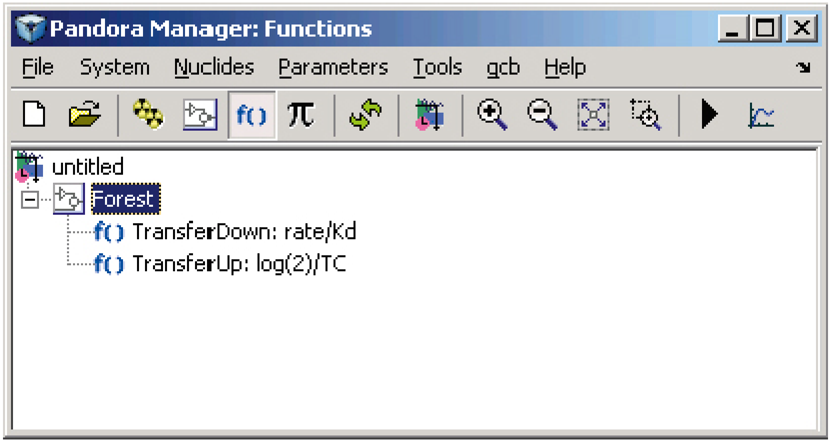
Figure 4-2. *Illustration of the Sub-system perspective.*

4.1.3 Function

In the Function perspective (f()), see Figure 4-3, all the functions in the model can be browsed and viewed. Here, the same concept as in the Sub-system perspective is applied: the system is browsed as a tree. However, only the Function blocks are shown here, and the leaves are the expressions defined in the blocks.

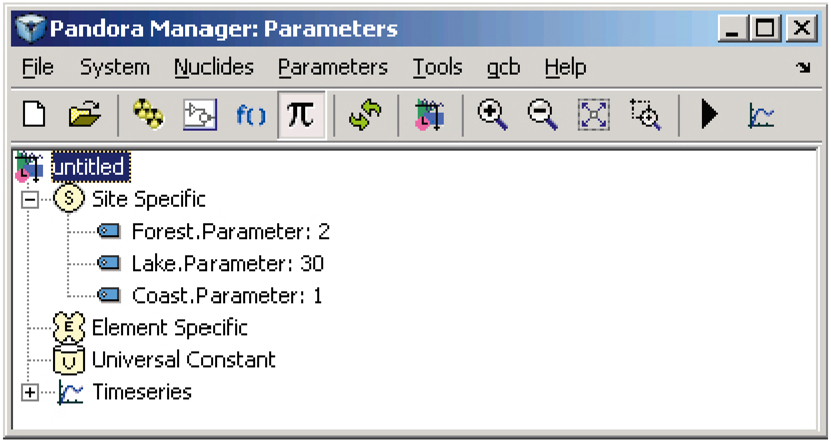
4.1.4 Parameter

In the Parameter perspective (π), see Figure 4-4, all the parameters and time-series in the model can be browsed and viewed. Here the same concept as in the Sub-system perspective is applied: the system is browed as a tree. However, only the parameters and time-series are shown. If there are variables with numerical values in the workspace these values will be shown as well, otherwise the leaves will show "not a number" (NaN).


Pandora Manager: Functions
File System Nuclides Parameters Tools gcb Help

untitled
Forest
f() TransferDown: rate/Kd
f() TransferUp: log(2)/TC

Figure 4-3. *Illustration of the Function perspective.*



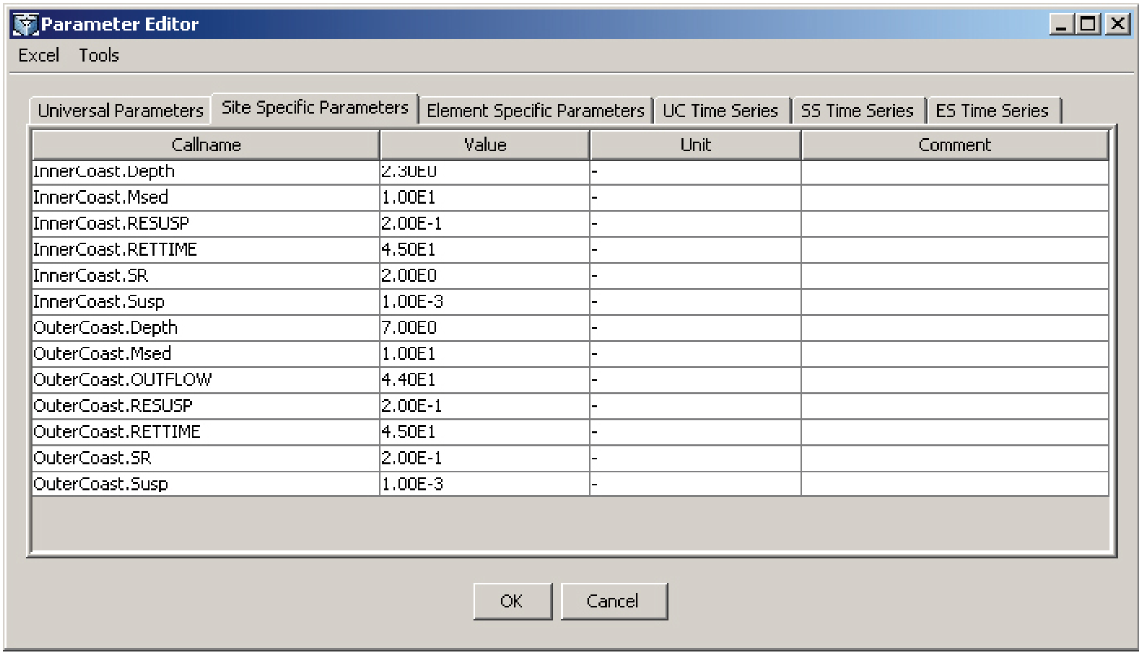
untitled
- Site Specific
  - Forest.Parameter: 2
  - Lake.Parameter: 30
  - Coast.Parameter: 1
- Element Specific
- Universal Constant
- Timeseries

Figure 4-4. *Illustration of the Parameter perspective.*

4.2 Parameter Manager

In Pandora, there is a concept to separate the model from input data. Therefore, no parameter values are stored inside the model; only links to data in MATLAB workspace are stored. To be able to make a simulation, these links must be properly connected to numeric input data. The Parameter Manager, see Figure 4-5, is a Java interface to aid the steps of numerical import of values into MATLAB workspace. It facilitates the editing of all model parameters from one central place. It is accessible from the "Parameters" menu in Pandora Manager.

When opening the Parameter Manager, Pandora browses the currently opened model for parameter and time-series blocks. These blocks are then stored in a parameter map together with comments and units defined inside the blocks as well as possible values previously defined in the MATLAB workspace. The parameter map is then sent to the Parameter Manager from which it is possible to update parameter values manually. From the Parameter Manager it is also possible to import and



Callname	Value	Unit	Comment
InnerCoast.Depth	2.30E0	-	
InnerCoast.Msed	1.00E1	-	
InnerCoast.RESUSP	2.00E-1	-	
InnerCoast.RETTIME	4.50E1	-	
InnerCoast.SR	2.00E0	-	
InnerCoast.Susp	1.00E-3	-	
OuterCoast.Depth	7.00E0	-	
OuterCoast.Msed	1.00E1	-	
OuterCoast.OUTFLOW	4.40E1	-	
OuterCoast.RESUSP	2.00E-1	-	
OuterCoast.RETTIME	4.50E1	-	
OuterCoast.SR	2.00E-1	-	
OuterCoast.Susp	1.00E-3	-	

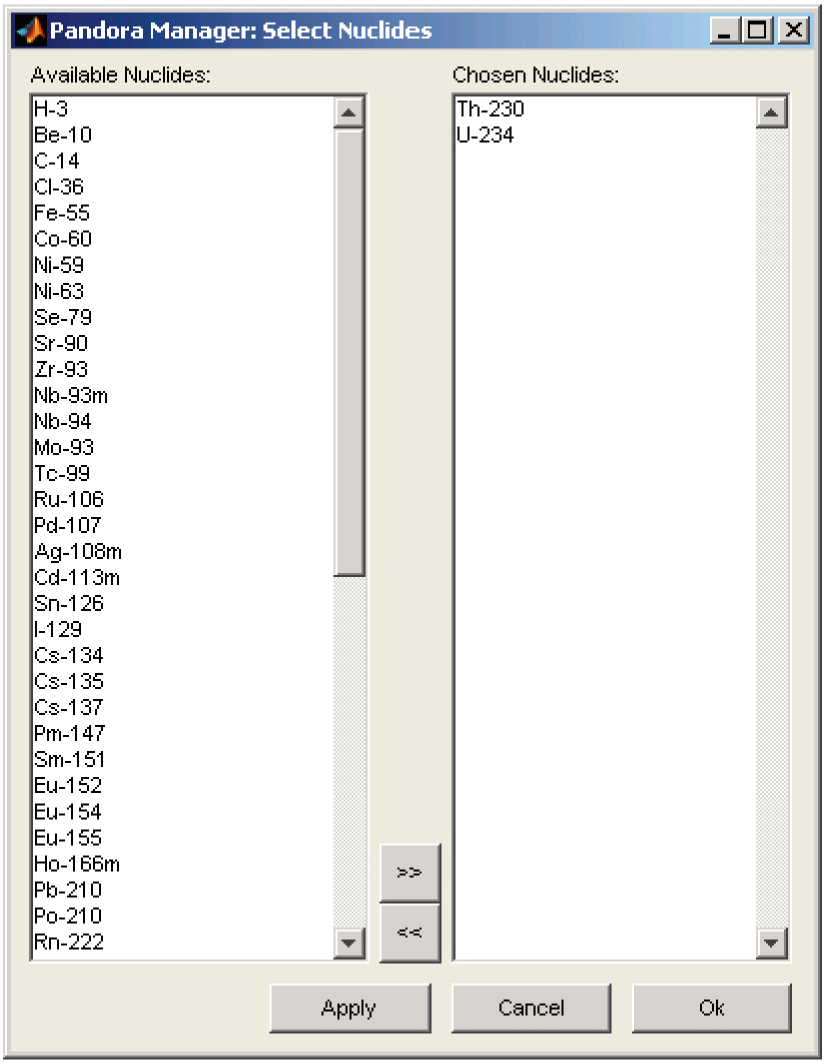
Figure 4-5. *Parameter Manager Interface.*

export parameter values from and to Excel sheets. When clicking on the OK button all values in the Value column are propagated into the correct place in the parameter and time-series structures in the MATLAB workspace, to be used when simulating the model. Changes in the Units and Comments field are not propagated into the model, so any changes that the user want to store have to be explicitly written in or exported to parameter Excel sheets.

4.3 Radionuclide selection

Under the Nuclides menu, there are several options to edit the modelled radionuclides within the model, i.e. the settings inside the Pandora block. Select Nuclides option allows the user to change elements in the Pandora block, possible elements comes from the nuclide database where radionuclide half-life is also stored. The select nuclides interface is illustrated in Figure 4-6.

Edit Decay Pairs option allows the user to change how in-growth of radionuclides is handled. In the interface, see Figure 4-7, decay pairs can be added and removed. A decay pair consists of a parent and a daughter radionuclide together with a branching ratio. The branching ratio is a number between 0 and 1 where 1 means that all decaying material of the parent radionuclide grows into the daughter radionuclide.



Pandora Manager: Select Nuclides

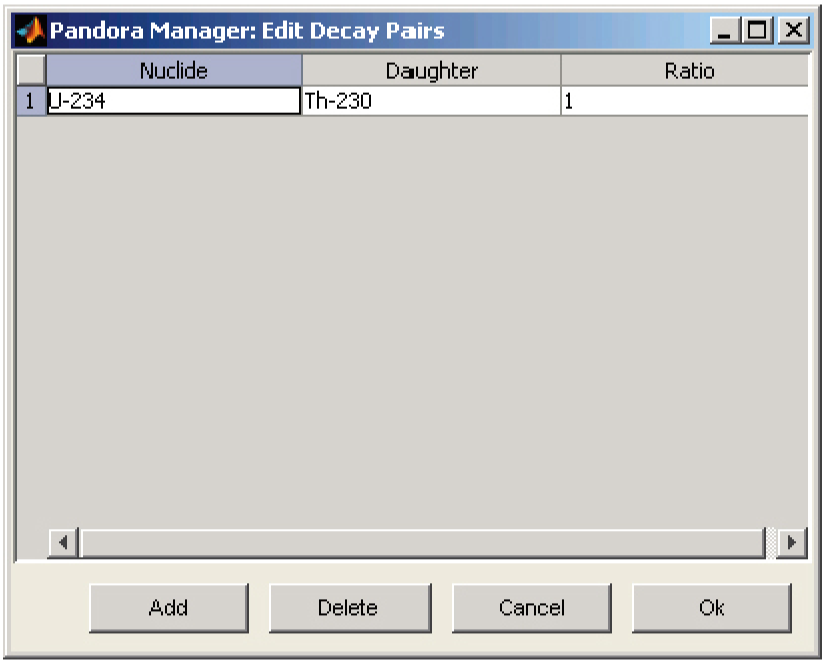
Available Nuclides:

H-3
Be-10
C-14
Cl-36
Fe-55
Co-60
Ni-59
Ni-63
Se-79
Sr-90
Zr-93
Nb-93m
Nb-94
Mo-93
Tc-99
Ru-106
Pd-107
Ag-108m
Cd-113m
Sn-126
I-129
Cs-134
Cs-135
Cs-137
Pm-147
Sm-151
Eu-152
Eu-154
Eu-155
Ho-166m
Pb-210
Po-210
Rn-222

Chosen Nuclides:

Th-230
U-234

Figure 4-6. *Select nuclides interface.*

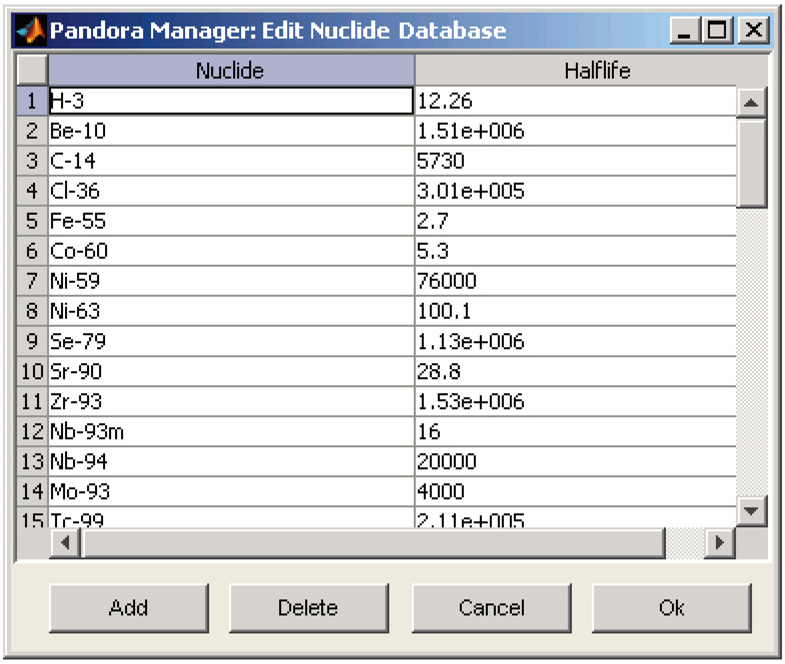


	Nuclide	Daughter	Ratio
1	U-234	Th-230	1

Figure 4-7. *Edit Decay Pairs interface.*

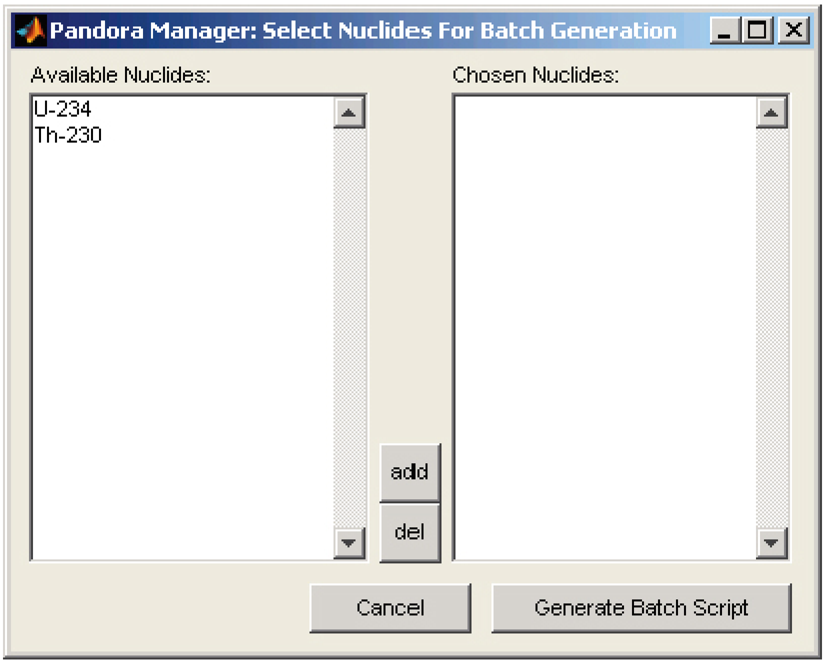
To edit the radionuclide database within Pandora, select the option Edit Nuclide Database. Figure 4-8 shows the interface to edit the radionuclide database.

Choose the Generate Batch Script option to open interface in Figure 4-9. This interface allows the user to generate a script that will run the model with different sets of nuclides.



	Nuclide	Halflife
1	H-3	12.26
2	Be-10	1.51e+006
3	C-14	5730
4	Cl-36	3.01e+005
5	Fe-55	2.7
6	Co-60	5.3
7	Ni-59	76000
8	Ni-63	100.1
9	Se-79	1.13e+006
10	Sr-90	28.8
11	Zr-93	1.53e+006
12	Nb-93m	16
13	Nb-94	20000
14	Mo-93	4000
15	Tc-99	2.11e+005

Figure 4-8. *Edit Nuclide Database interface.*



Pandora Manager: Select Nuclides For Batch Generation

Available Nuclides:

U-234
Th-230

Chosen Nuclides:

Figure 4-9. *Generate Batch Script interface.*

4.4 Verification of model connections

In Simulink it is relatively easy to make mistakes whilst adding and connecting different blocks with each other. Pandora has a verification feature intended to eliminate many of these mistakes. If one follows the rule that only Pandora blocks can be connected to Reservoir, Transfer and Function blocks, the verification feature can check that no errors have been made in these connections. To be able to open a Pandora model in Pandas (see Chapter 5) the model has to pass the verification call. In addition, when generating a report (see Section 4.3) it is preferable (but not required) to have a verified model.

4.5 Ghosting Reservoir and Transfer blocks

Using the Ghost button (👻) in the Pandora Manager panel, see Figure 4-5, one can create so called ghost copies of the currently selected Transfer or Reservoir block. A ghost copy is a similar looking block with a built-in From-block that refers to the outgoing signal of the referred block. Use this feature if there is a need to use the magnitude of the block somewhere else in the same system but do not want to have a line between the blocks in the model. Important to note is that the ghost copy can only be used within the same subsystem as the block it represents. Furthermore, renaming of the block is not propagated to the ghost copy, one need to manually rename the ghost or create a new copy.

4.6 Report generation

A Microsoft Word document documenting the Pandora model can be generated from the 'Tools|Create Report…' menu in Pandora Manager. The report will consist of a title page, a table of contents, an introduction, a section with model settings, a section about the nuclides in the model, a section about the outputs in the model and finally a section about each subsystem where there resides any Pandora blocks in the model. Each of these subsystem sections will have information about parameter and function blocks, their connections between each other as well as eventual differential equations building up the dynamic system. Figure 4-10 shows some snapshot examples of an automatically generated report of a Pandora model.

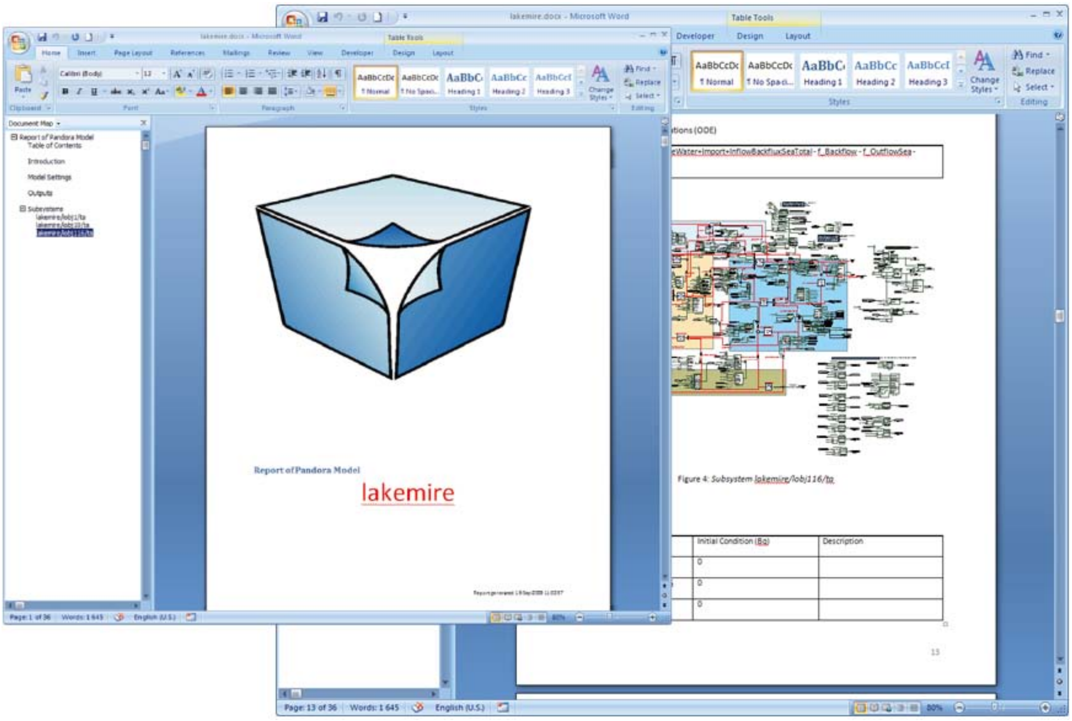


Figure 4-10. *Example of exported Word document.*

4.7 Variable search

Finding Simulink blocks using the “find_system” function from the MATLAB prompt is very good whilst looking for any type of Simulink block in a Simulink model. From the function manager there is a specialised Pandora-block search function, which will create direct links from the search results in the MATLAB prompt to specified blocks in the Pandora model. All Pandora block names in the model matching the search text using regular expressions are listed as hyperlinks in the MATLAB prompt.

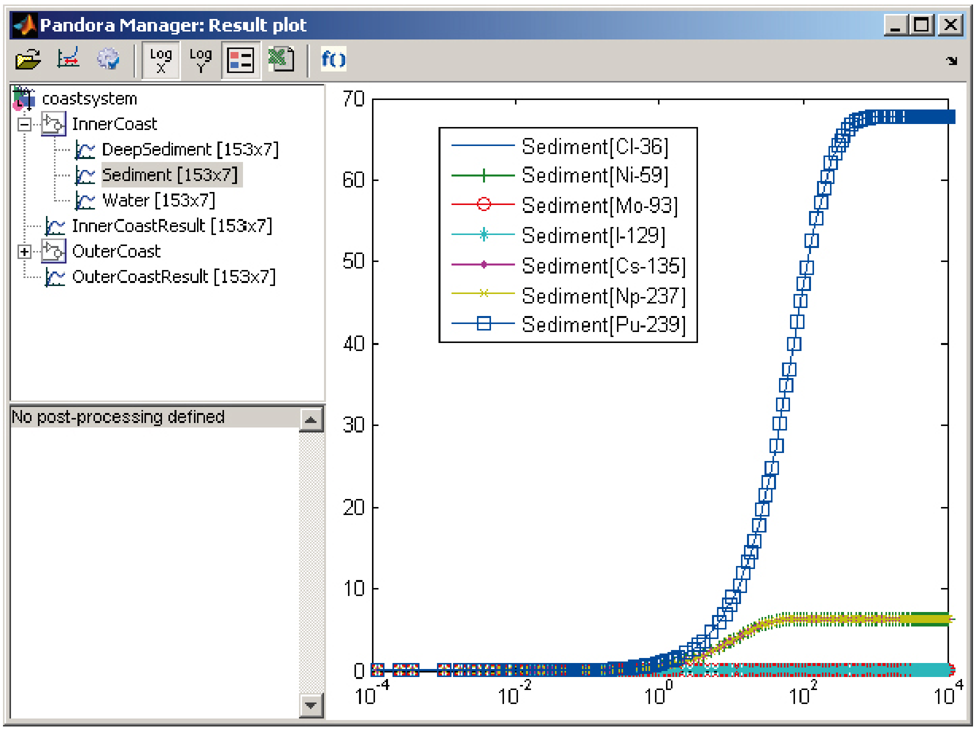
4.8 Result viewer

The Result Viewer, see Figure 4-11, is a tool to review, analyse and export results from a model simulation. Access the tool by clicking the Result Viewer push button in the Pandora Manager Toolbar (). Another way to reach the tool is by double-clicking on a Result block. Variables that can be analysed using the Result Viewer are values coming from Reservoir blocks and Result blocks. For Reservoir blocks, the check box for data logging of the signal needs to be checked for the data to be saved. In addition, the Signal logging check box in the models ‘Configuration parameters…’-menu (under Data Import/Export) also has to be checked, see Figure 4-12.

After a simulation a Simulink logging variable has been saved to the MATLAB workspace with the name defined in the text-field next to the Signal logging check box in ‘Simulation|Configuration Parameters…’ menu in Simulink, as seen in Figure 4-12 (for example logsout). This variable stores all values from the simulation and can be loaded into the Result Viewer. Use the Open push button () in the Result Viewer toolbar to load the Signal logging data structure into the tool. When opening a new session of Result Viewer, either from the toolbar or by double-click on a Result block in the currently simulated model the new session of Result Viewer tries to load the defined Signal logging structure automatically.

Post-processing variables can be generated and analysed within the Result Viewer if defined in the model using Post-processing blocks (see Section 3.5.5).

From the Result Viewer it is possible to export the currently viewed data to either an external MATLAB figure () or to an Excel data figure () from which there are possibilities to obtain time specific data by interpolating time points.

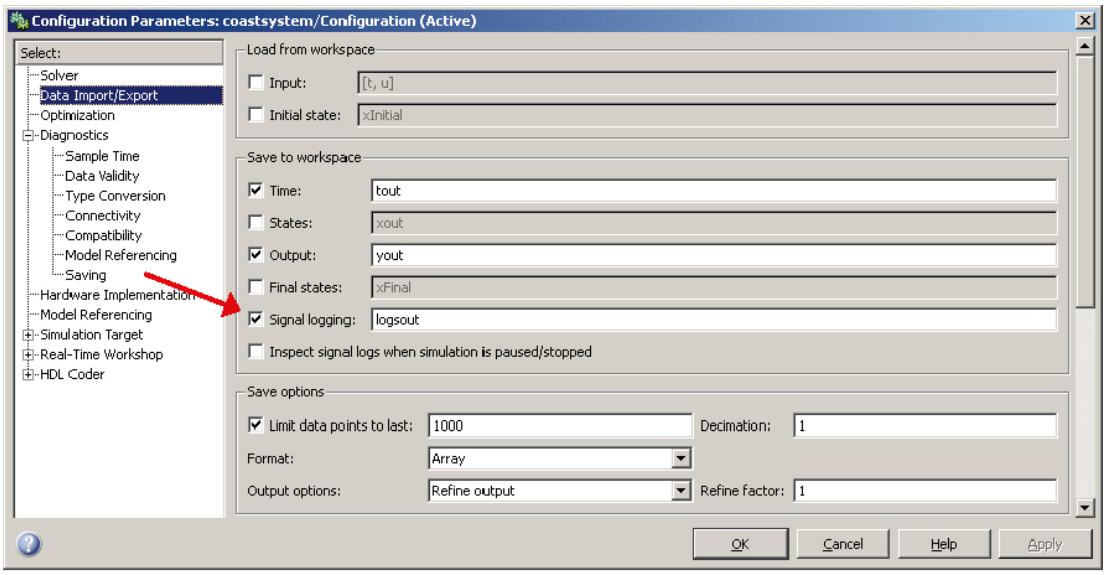


Pandora Manager: Result plot

- coastsystem
  - InnerCoast
    - DeepSediment [153x7]
    - Sediment [153x7]
    - Water [153x7]
  - InnerCoastResult [153x7]
  - OuterCoast
  - OuterCoastResult [153x7]

No post-processing defined

Figure 4-11. *Result viewer Interface.*



Configuration Parameters: coastsystem/Configuration (Active)

Select:
- Solver
- Data Import/Export
- Optimization
- Diagnostics
  - Sample Time
  - Data Validity
  - Type Conversion
  - Connectivity
  - Compatibility
  - Model Referencing
  - Saving
- Hardware Implementation
- Model Referencing
- Simulation Target
- Real-Time Workshop
- HDL Coder

Load from workspace

□ Input: [t, u]

□ Initial state: xInitial

Save to workspace

☑ Time: tout

□ States: xout

☑ Output: yout

□ Final states: xFinal

☑ Signal logging: logsout

□ Inspect signal logs when simulation is paused/stopped

Save options

☑ Limit data points to last: 1000 Decimation: 1

Format: Array

Output options: Refine output Refine factor: 1

Figure 4-12. *Make sure Signal logging is checked.*

4.9 Get Current Block

Under the get current block (GCB) menu there are options for adding and connecting blocks to the in-ports and out-ports on the current selected Pandora block. If no Pandora block is selected the GCB-menu does not do anything. If the current selected block has unconnected in-ports, a submenu called ‘Inport’ will be created. Under this submenu, there is a list of the unconnected in-ports. For each of the in-ports one can choose to connect automatically any of Universal Constant, Element Specific, Site Specific, Time Series, Interface, Constant, Ground or Inport blocks. One can also choose to connect automatically the same type of block to all unconnected in-ports. The same goes for the current block out-port. However, for out-ports it is always possible to connect new blocks. The types of blocks to connect to out-port are Plot, Result, Outport and Terminate blocks.

Figure 4-13 illustrates the usage of the GCB menu. Having a Function block one may want to connect Element Specific constant blocks as input and an Outport as output. Using the GCB menu one can automatically create these blocks, name and connect them correctly directly from the menu instead of drag-and-drop from the Pandora Library, rename and connect them manually.

4.10 Function Manager

Function Manager changes the input interface of the Function block and provides an alternative way of working with the block. By default, the function block works such that the in-ports of the block generate automatically, defined by the unknown variables in the equation of the block. Usually the user start with a block without any in-ports, then after adding the equation, the in-ports for the unknown variables appear and the next step is to connect corresponding input variables to the automatically generated in-ports. Whilst using the Function Manager, instead of the standard mask dialog a more advanced MATLAB user interface will appear after double-click on the function block. To enable the Function Manager, check the field in the Pandora Manager: Tools|Function Manager.

All connected Pandora blocks will appear as input variables inside the block and the user can simply define the equation by point and click with the mouse. If one wants to connect more input variables, one has to add more in-ports and connect Pandora blocks to these. Automatically the in-ports will be listed in the interface with their corresponding Pandora names. The workflow using the Function Manager is therefore different from the standard way of first defining the equation and then adding the input variables. Using the Function Manager, you preferably begin with defining the input variables to be used by the function before defining the equation.

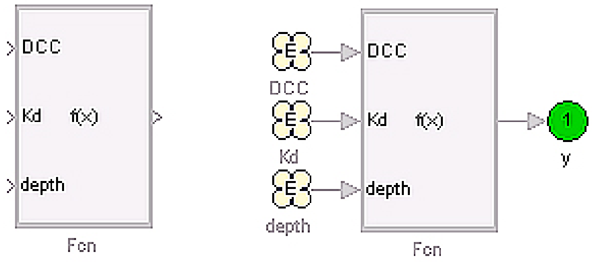


Figure 4-13. *Using GCB to quickly connect Pandora blocks to a Function block.*

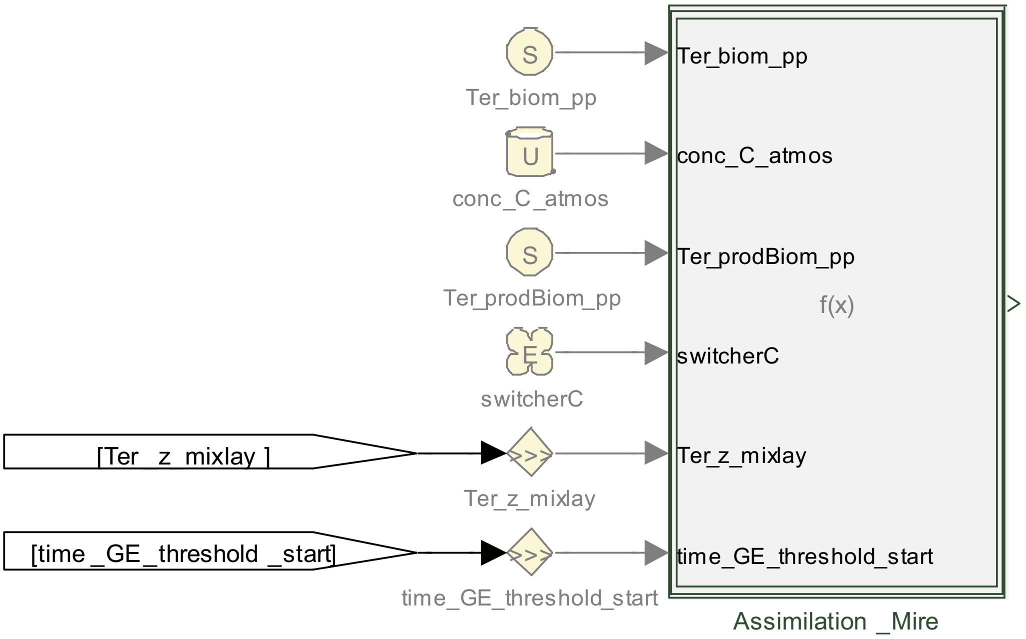
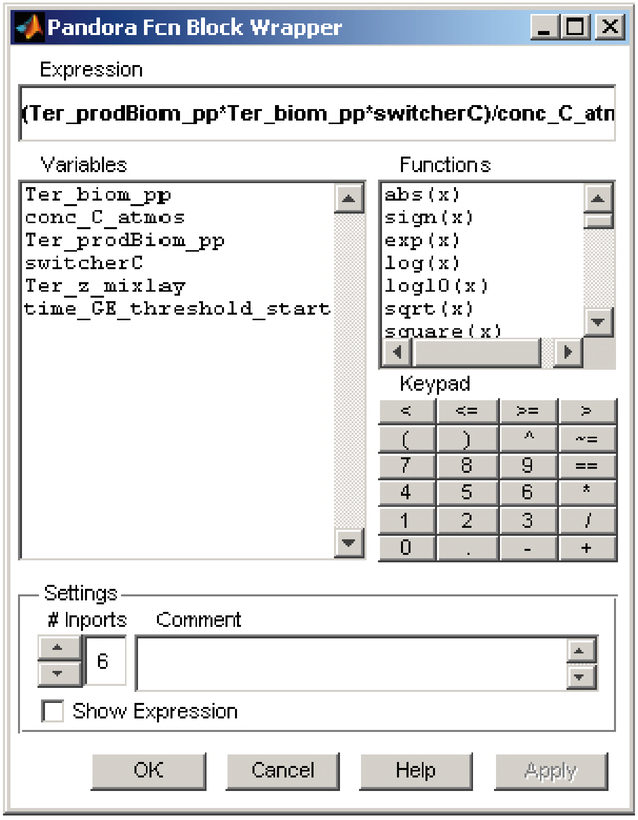


Figure 4-14. *Example of Function block with connected input variables.*

Figure 4-14 shows an example of connected Pandora blocks to a Function block, Figure 4-15 and Figure 4-16 illustrates the corresponding interfaces for modifying the block after a double-click on the block using Function Manager and not using it respectively.



Pandora Fcn Block Wrapper

Expression

(Ter_prodBiom_pp*Ter_biom_pp*switcherC)/conc_C_atn

Variables

```
Ter_biom_pp
conc_C_atmos
Ter_prodBiom_pp
switcherC
Ter_z_mixlay
time_GE_threshold_start
```

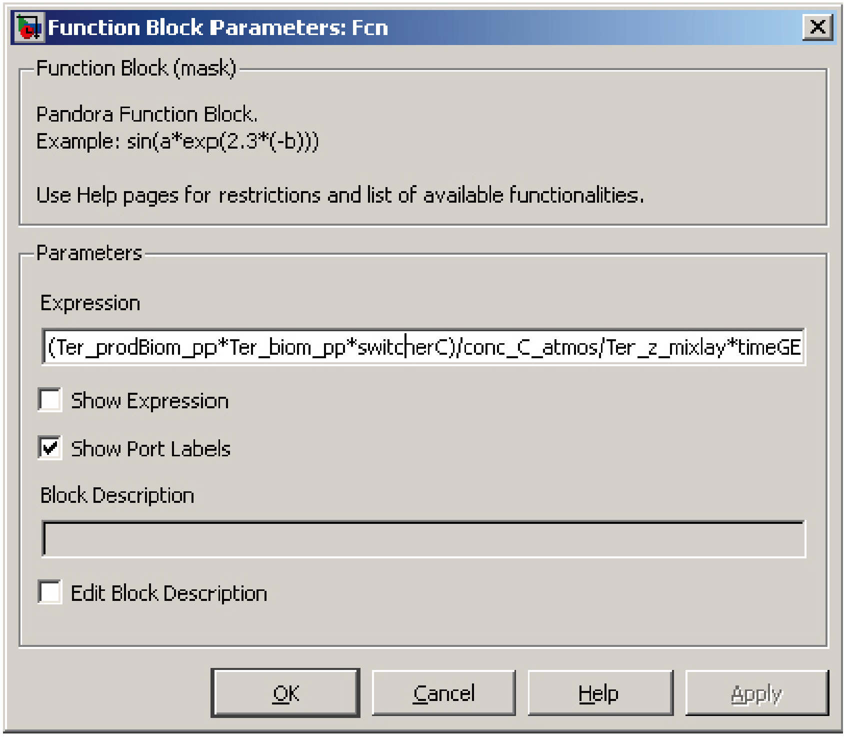
Settings

# Inports: 6

Comment:

☐ Show Expression

Figure 4-15. *Function Manager interface corresponding to Function block in Figure 4-14.*



Function Block Parameters: Fcn

Function Block (mask)

Pandora Function Block.
Example: sin(a*exp(2.3*(-b)))

Use Help pages for restrictions and list of available functionalities.

Parameters

Expression

(Ter_prodBiom_pp*Ter_biom_pp*switcherC)/conc_C_atmos/Ter_z_mixlay*timeGE

☐ Show Expression

☑ Show Port Labels

Block Description

☐ Edit Block Description

Figure 4-16. *Mask dialog corresponding to Function block in Figure 4-14.*

5 Pandas

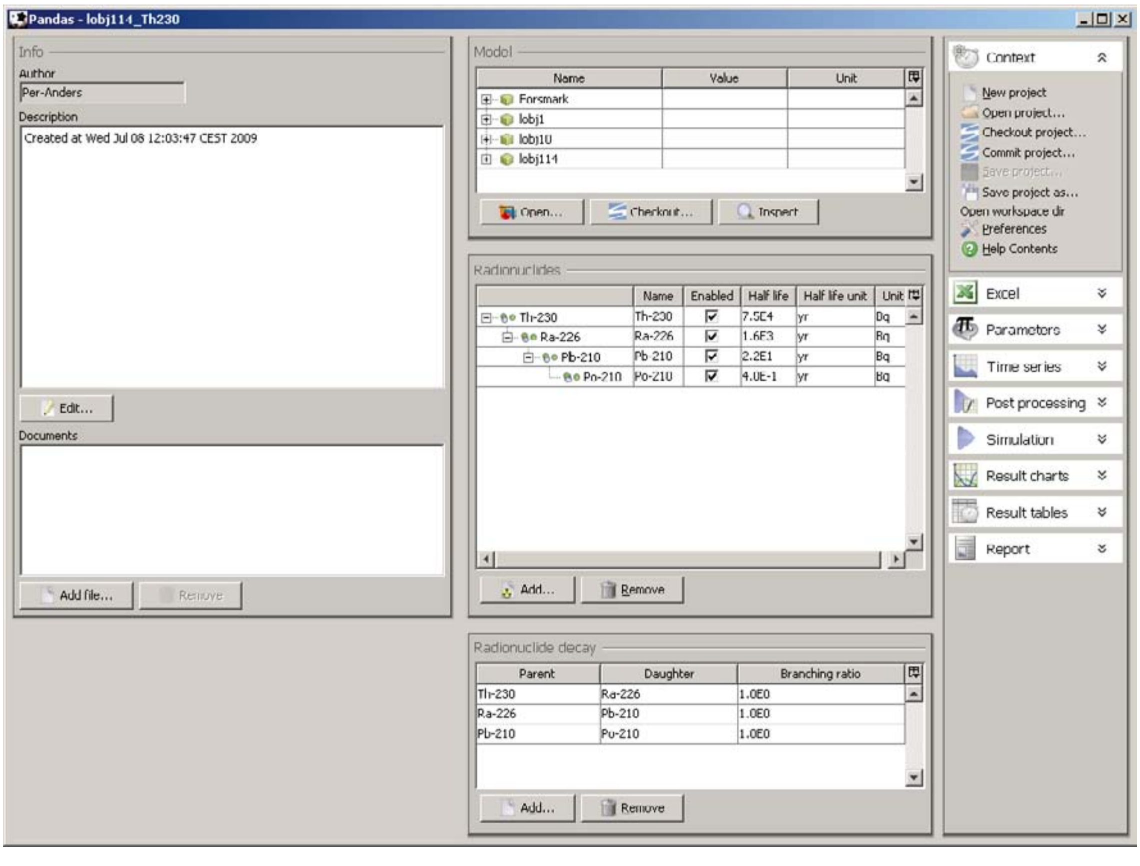
Pandas (PANDora ASsessment) is a Java based software developed for creating and working with assessments using Pandora models and Excel parameter sheets as input data. An assessment consists of a Pandora model, a radionuclide chain, values for parameter and time-series (by their link to Excel files), post-processing functions, simulation settings, result charts and result tables as well as a report and imported documents and added comments. The result data in an assessment can either be deterministic, a batch simulation with parameter data drawn from a simulation table, a probabilistic Monte Carlo simulation or a sensitivity analysis.

The incentive for developing a Java-based tool to handle assessments was to obtain a consistent version-handling strategy, to have a standardised and guaranteed connection between input data, the model and the obtained results, and to standardise the assessment creation procedure. Pandas supports sub-version handling of input files to avoid uncertainty about which version currently is being used. Since the tool packages the data together with model and results the connection between these are set, eliminating any potential confusion between different input data sets.

The Pandas interface is divided into nine different software perspectives. These perspectives are described in the following sub-sections.

5.1 Context

The Context perspective, see Figure 5-1, is the first view presented when starting Pandas. This is where you will manage your assessment project. From the rightmost menu buttons, you can create a new, open a previously defined, check out, commit or save an assessment project. From the Preferences menu button, you can change the definitions for default subversion directory, username and password.



Pandas - lobj114_Th230

Info

Author: Per-Anders

Description: Created at Wed Jul 08 12:03:47 CEST 2009

Documents

Model

Name	Value	Unit
Forsmark		
lobj1		
lobj10		
lobj114		

Radionuclides

	Name	Enabled	Half life	Half life unit	Unit
Th-230	Th-230	☑	7.5E4	yr	Bq
Ra-226	Ra-226	☑	1.6E3	yr	Bq
Pb-210	Pb-210	☑	2.2E1	yr	Bq
Po-210	Po-210	☑	4.0E-1	yr	Bq

Radionuclide decay

Parent	Daughter	Branching ratio
Th-230	Ra-226	1.0E0
Ra-226	Pb-210	1.0E0
Pb-210	Po-210	1.0E0

Figure 5-1. Pandas Context perspective.

For each new assessment project, you need to add a Pandora model. This is done from the Model window and you can either open a model from your local computer or obtain it by checking out the model from sub-version using the Checkout button. The Pandora model blocks are shown inside the tree model in the Model window. Inside the Pandora model there is always a Pandora block, from this the radionuclide information is extracted and propagated into Pandas Radionuclides window and Radionuclide decay window. The radionuclide data can later be modified from these windows. In the two leftmost windows in the Context perspective, information about the project can be added and there is a possibility to add documents to the project.

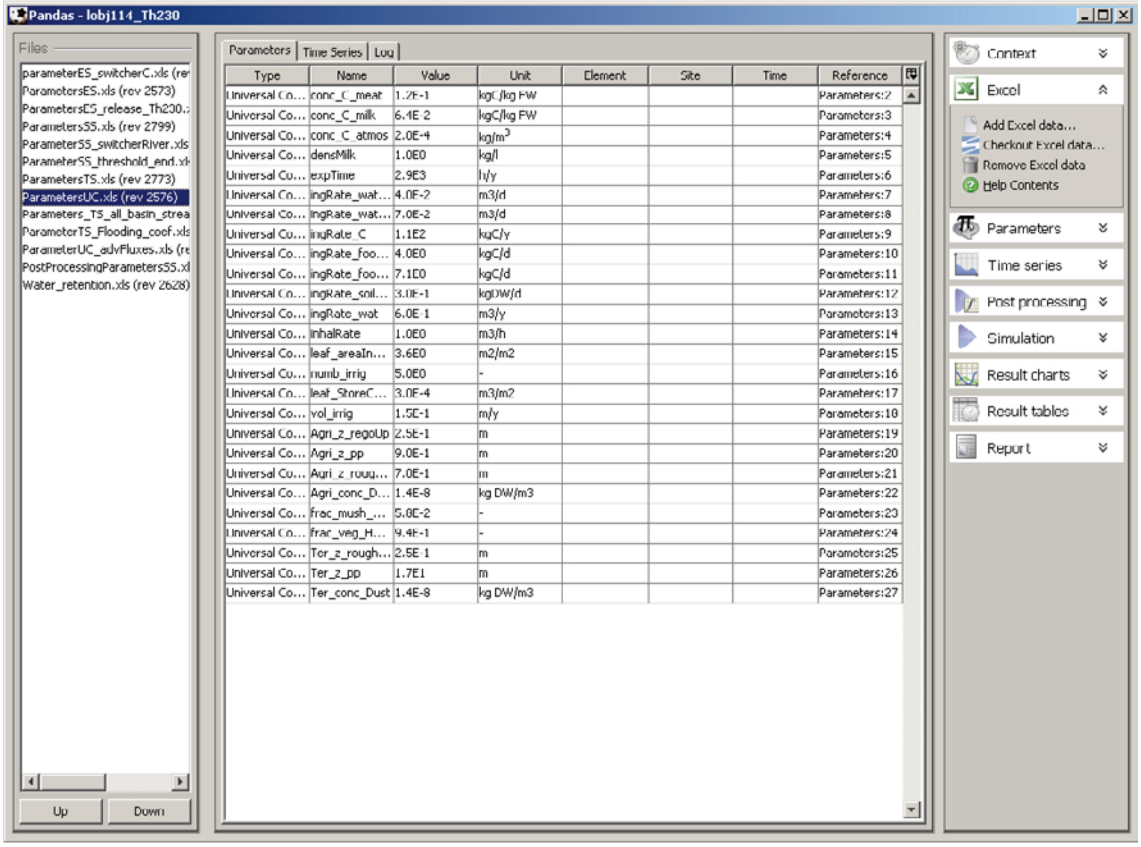
5.2 Excel

In the Excel perspective, see Figure 5-2, one manages parameter in-data files. You can add Excel sheets from your local computer system or “check out” from a Sub-version server. If data comes from a version handled sub-version file this is logged in the assessment. Since the same parameter values can exist in several parameter files, the parameter value that is found first is used. To order the Excel-sheets, use the up and down arrows in the lower left corner of the window. Excel sheets are read in a top-down approach.

There is no problem if Excel sheet has parameter values defined that do not have a matching parameter block in the Pandora model. These values are read but ignored in the assessment.

5.2.1 Parameter format

Parameter values used in Pandora models are stored in Excel files with a specific formatting. The easiest ways to create these files are by exporting parameters from the model using either the Pandora Manager or Pandas. A recommendation is to separate the different kinds of parameters into different files but that is not a requirement.



parameterES_switcherC.xls (re
ParametersES.xls (rev 2573)
ParametersES_release_Th230.:
ParametersSS.xls (rev 2799)
ParameterSS_switcherRiver.xls
ParameterSS_threshold_end.xl
ParametersTS.xls (rev 2773)
ParametersUC.xls (rev 2576)
Parameters_TS_all_basin_strea
ParameterTS_Flooding_coef.xls
ParameterUC_advFluxes.xls (re
PostProcessingParametersSS.xl
Water_retention.xls (rev 2628)

Type	Name	Value	Unit	Element	Site	Time	Reference
Universal Co...	conc_C_meat	1.2E-1	kgC/kg FW				Parameters:2
Universal Co...	conc_C_milk	6.4E-2	kgC/kg FW				Parameters:3
Universal Co...	conc_C_atmos	2.0E-4	kg/m^3				Parameters:4
Universal Co...	densMilk	1.0E0	kg/l				Parameters:5
Universal Co...	expTime	2.9E3	h/y				Parameters:6
Universal Co...	ingRate_wat...	4.0E-2	m3/d				Parameters:7
Universal Co...	ingRate_wat...	7.0E-2	m3/d				Parameters:8
Universal Co...	ingRate_C	1.1E2	kgC/y				Parameters:9
Universal Co...	ingRate_foo...	4.0E0	kgC/d				Parameters:10
Universal Co...	ingRate_foo...	7.1E0	kgC/d				Parameters:11
Universal Co...	ingRate_soil...	3.0E-1	kgDW/d				Parameters:12
Universal Co...	ingRate_wat	6.0E-1	m3/y				Parameters:13
Universal Co...	inhalRate	1.0E0	m3/h				Parameters:14
Universal Co...	leaf_areaIn...	3.6E0	m2/m2				Parameters:15
Universal Co...	numb_irrig	5.0E0	-				Parameters:16
Universal Co...	leaf_StoreC...	3.0E-4	m3/m2				Parameters:17
Universal Co...	vol_irrig	1.5E-1	m/y				Parameters:18
Universal Co...	Agri_z_regoUp	2.5E-1	m				Parameters:19
Universal Co...	Agri_z_pp	9.0E-1	m				Parameters:20
Universal Co...	Agri_z_roug...	7.0E-1	m				Parameters:21
Universal Co...	Agri_conc_D...	1.4E-8	kg DW/m3				Parameters:22
Universal Co...	frac_mush_...	5.0E-2	-				Parameters:23
Universal Co...	frac_veg_H...	9.4E-1	-				Parameters:24
Universal Co...	Ter_z_rough...	2.5E-1	m				Parameters:25
Universal Co...	Ter_z_pp	1.7E1	m				Parameters:26
Universal Co...	Ter_conc_Dust	1.4E-8	kg DW/m3				Parameters:27

Figure 5-2. Pandas Excel perspective.

Each Excel file should begin with a header section at the first two rows followed by the parameter definitions. Each row in the Excel sheet then represents a unique parameter value. If the same unique definition occurs more than once the first occurrence is used at importation stage. The header is separated by two different sections; one is dependencies and the other is the parameter values. This division in sections should be at the first row in the Excel sheet and say "Dependencies" above the dependencies section and "Parameters" above the parameter values section. Table 5-1 illustrates how the Pandora Excel file look like if exported from Pandas or the parameter manager.

Table 5-1. Example illustration of Excel format.

1	**Dependencies**			**Parameter**									
2	Site	Nuclide	Time	Name	Type	Value	Unit	Mean	Std. Dev.	Min	Max	Distribution	Comment
3 …													

Different types of parameters require different fields in the Dependencies section to be filled in. The field Type in the Parameter section defines the type of the parameter. Following are the different requirements for the different parameter types:

- Site-specific: Site should define the subsystem path to the location of the parameter; subsystems are separated by a dot (.). Type is SS.
- Universal constant: has no dependency. Type is UC.
- Element specific: Nuclide should define the radionuclide or element of the parameter value. Type is ES.
- Time-series: Site and Time must be defined. Site is the same as for Site-specific parameters and Time is the specific time-point the value is defined for. Type is SSTS.

5.3 Parameters

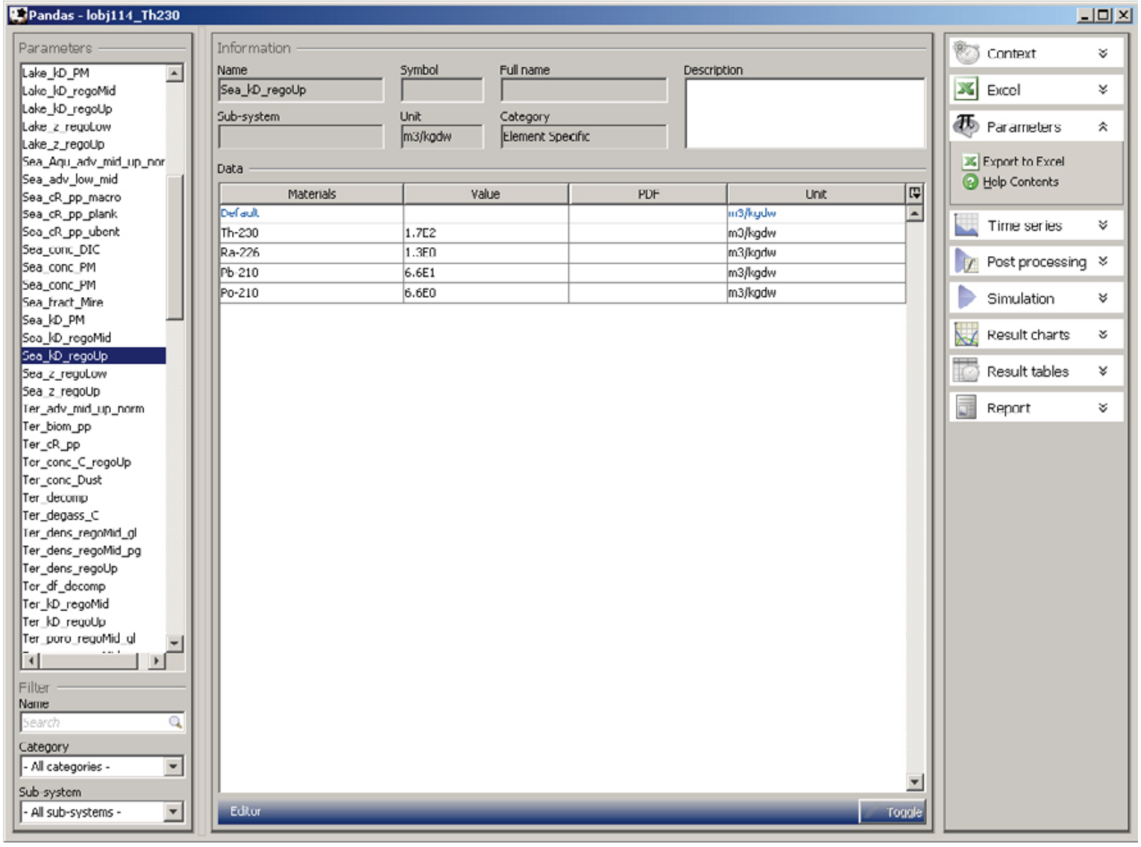
In the Parameters perspective, see Figure 5-3, all parameter blocks used in the Pandora model are listed. Parameters that do not exist in model but in the imported Excel sheets are ignored. In the lower left corner, there are filtering possibilities for easier management of large parameter sets. Exporting parameters to an external Excel sheet is done by selecting them and clicking the button "Export to Excel".

Values and information about the parameter are read from parameter Excel sheets. It is also possible to add parameter values manually; these values are marked with a bold font and listed as unsecure non-versioned values. In the Reference column the exact position inside the imported Excel files from where the parameter data has been extracted are listed.

5.4 Time series

In the Time series perspective, see Figure 5-4, all Time-series blocks that exist in the Pandora model are listed. Time-series that do not exist in model but in imported Excel sheets are ignored. In the lower left corner, there are filtering possibilities as in the Parameter Perspective for easier management of large Time-series sets. Exporting Time-series to an external Excel sheet is done by selecting them and clicking the button "Export to Excel".

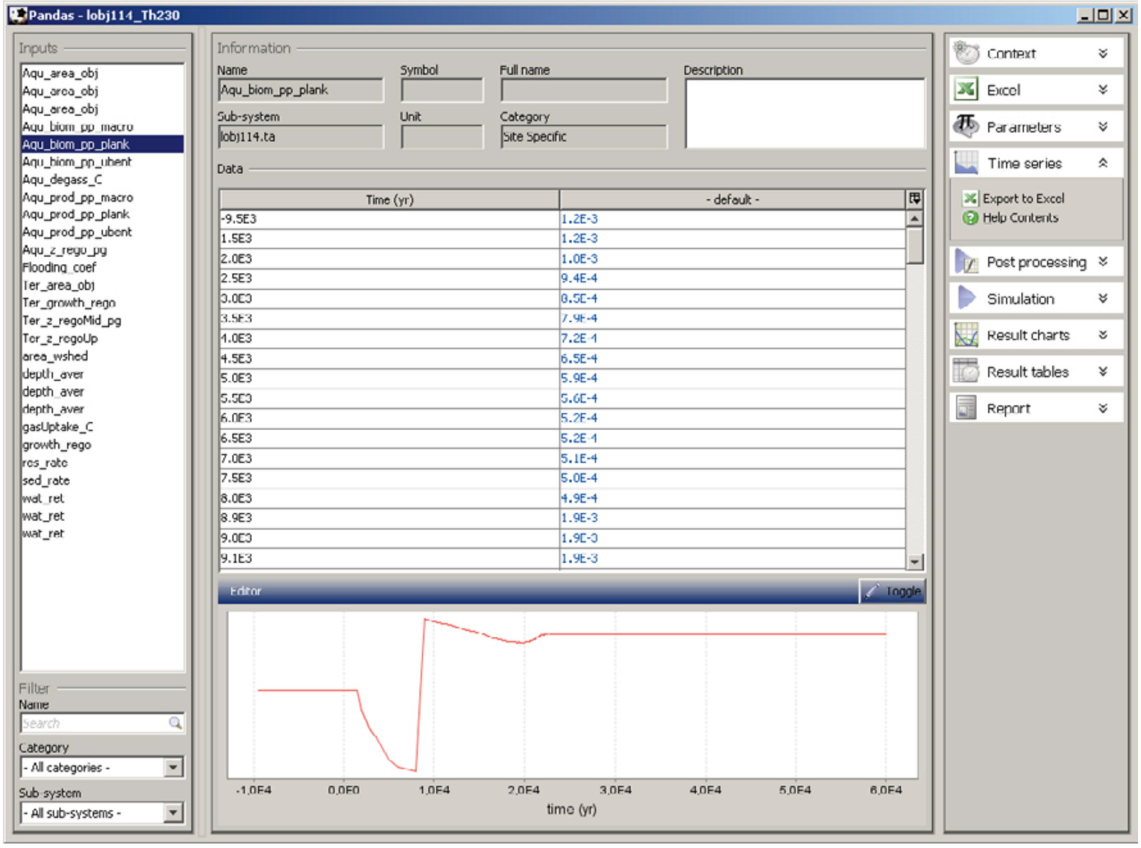
Values and information about the time series are read from Excel sheets. A graph illustrating interpolated values is shown underneath the input dialog.



Name: Sea_kD_regoUp
Symbol:
Full name:
Description:
Sub-system:
Unit: m3/kgdw
Category: Element Specific

Materials	Value	PDF	Unit
Default			m3/kgdw
Th-230	1.7E2		m3/kgdw
Ra-226	1.3E0		m3/kgdw
Pb-210	6.6E1		m3/kgdw
Po-210	6.6E0		m3/kgdw

***Figure 5-3.** Pandas Parameters perspective.*



Name: Aqu_biom_pp_plank
Symbol:
Full name:
Description:
Sub-system: lobj114.ta
Unit:
Category: Site Specific

Time (yr)	- default -
-9.5E3	1.2E-3
1.5E3	1.2E-3
2.0E3	1.0E-3
2.5E3	9.4E-4
3.0E3	8.5E-4
3.5E3	7.9E-4
4.0E3	7.2E-4
4.5E3	6.5E-4
5.0E3	5.9E-4
5.5E3	5.6E-4
6.0E3	5.2E-4
6.5E3	5.2E-4
7.0E3	5.1E-4
7.5E3	5.0E-4
8.0E3	4.9E-4
8.9E3	1.9E-3
9.0E3	1.9E-3
9.1E3	1.9E-3

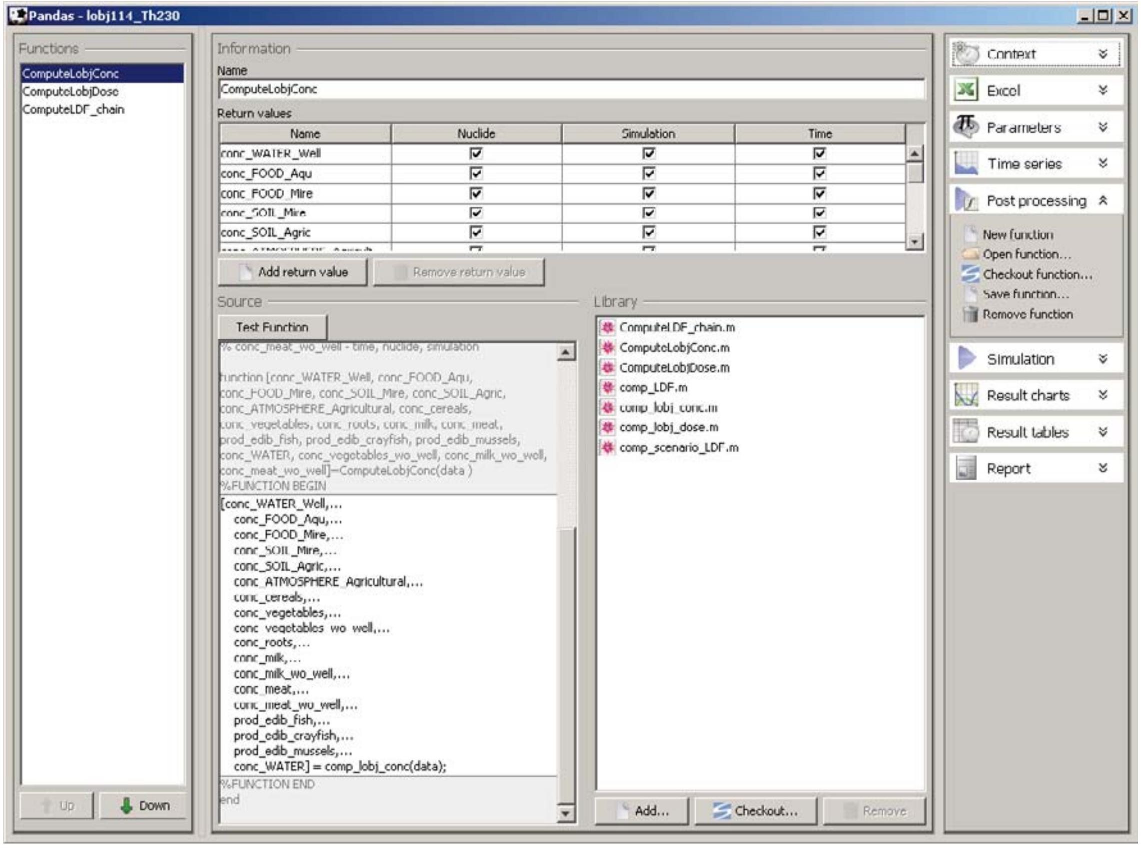
***Figure 5-4.** Pandas Time series perspective.*

5.5 Post processing

In the Post processing perspective, one manages all post-processing functions. A post-processing function is a MATLAB function that is evaluated after all dynamic Simulink model simulations are performed. Each post-processing function can return any number of resulting variables. You need to create and name these result-variables and define their dimension. A wrapper MATLAB function is automatically generated for each post-processing function as can be seen in Figure 5-5. The white area in this function is editable and is where you define your equations for the return variables. If your function is very complex or large you can preferably write a function in MATLAB and then import (check out) it to the library and then just call it from the function wrapper.

All output data chosen in Simulation outputs in Simulation settings (see Section 5.6) will be accessible in the data structure that is input argument to the wrapper function. Variables inside subsystems will be accessible using dot-notation. Three extra variables will also be accessible: the number of modelled nuclides (nnuc), the number of time points (ntim) and the number of iterations (niter).

Post processing functions will be evaluated in top-down order and you will therefore have access to variables computed in all previous post-processing functions in the current function. It is important that all returned output variables have the correct defined dimensions.



Name: ComputeLobjConc

Return values

Name	Nuclide	Simulation	Time
conc_WATER_Well	☑	☑	☑
conc_FOOD_Aqu	☑	☑	☑
conc_FOOD_Mire	☑	☑	☑
conc_SOIL_Mire	☑	☑	☑
conc_SOIL_Agric	☑	☑	☑

Add return value   Remove return value

Test Function

```
% conc_meat_wo_well - time, nuclide, simulation

function [conc_WATER_Well, conc_FOOD_Aqu,
conc_FOOD_Mire, conc_SOIL_Mire, conc_SOIL_Agric,
conc_ATMOSPHERE_Agricultural, conc_cereals,
conc_vegetables, conc_roots, conc_milk, conc_meat,
prod_edib_fish, prod_edib_crayfish, prod_edib_mussels,
conc_WATER, conc_vegetables_wo_well, conc_milk_wo_well,
conc_meat_wo_well]=ComputeLobjConc(data )
%FUNCTION BEGIN
[conc_WATER_Well,...
   conc_FOOD_Aqu,...
   conc_FOOD_Mire,...
   conc_SOIL_Mire,...
   conc_SOIL_Agric,...
   conc_ATMOSPHERE_Agricultural,...
   conc_cereals,...
   conc_vegetables,...
   conc_vegetables_wo_well,...
   conc_roots,...
   conc_milk,...
   conc_milk_wo_well,...
   conc_meat,...
   conc_meat_wo_well,...
   prod_edib_fish,...
   prod_edib_crayfish,...
   prod_edib_mussels,...
   conc_WATER] = comp_lobj_conc(data);
%FUNCTION END
end
```

Figure 5-5. Pandas Post processing perspective.

5.6 Simulation

From the Simulation perspective, see Figure 5-6, the model simulation settings are editable. For example; the simulation start and end time, the type of simulation and the settings for different simulation types. The four different simulation types are:

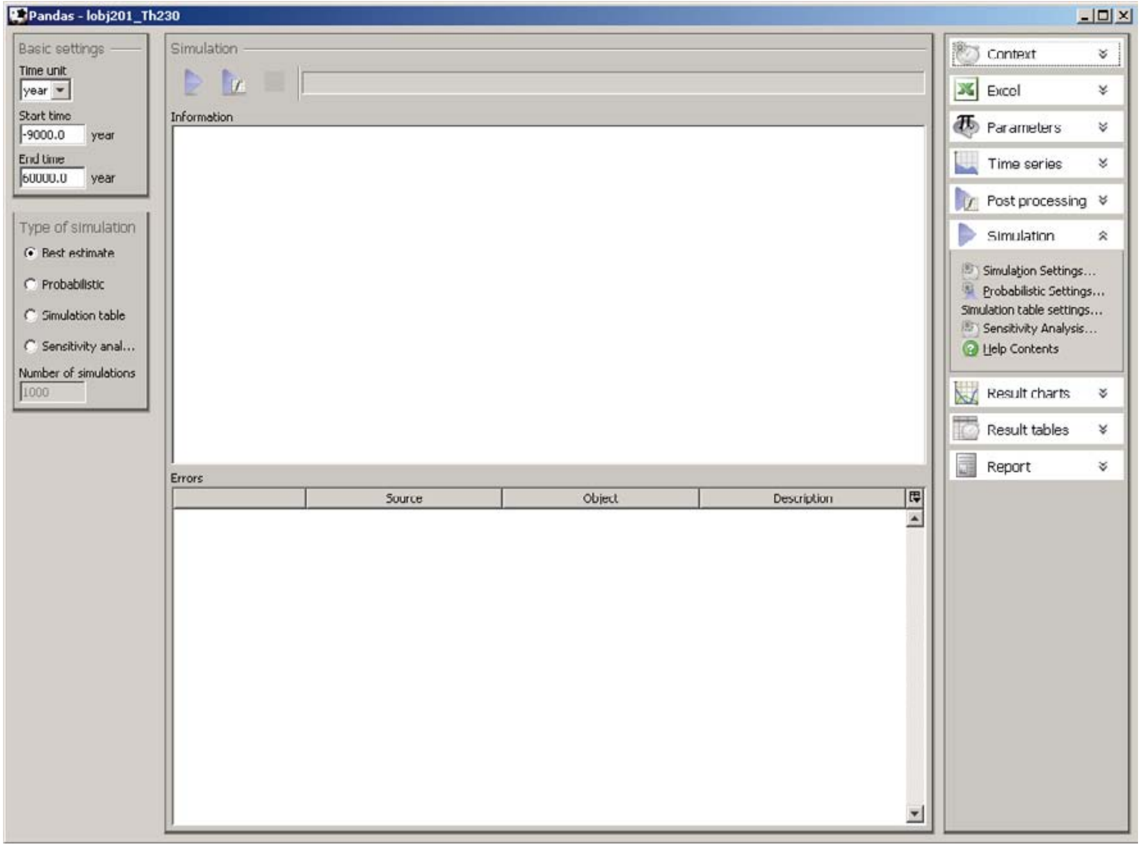
1. Best estimate – deterministic simulation with best-estimated values used for each parameter.
2. Probabilistic – probabilistic simulation where parameters can either be chosen to use their best-estimated values or are drawn from defined probability density functions.
3. Simulation table – a set of simulations where chosen parameters can be changed in turn, rest of parameters uses their best-estimated values.
4. Sensitivity analysis – different sensitivity analysis methods can be chosen.

In the simulation perspective, you also start the simulation, simply by clicking the simulation button. Post-processing is performed by clicking the post-processing button, after the simulation has finished. Below the Information window there is a window displaying eventual errors. An example of a possible error is missing or erroneous parameter values. All errors have to be solved before running a simulation.

In the following sub-sections, settings for the different simulation types are described.

5.6.1 Simulation Settings

The simulation settings window contains three pages: General, Outputs and Solver. In the General page you can edit the simulation interval as well as which time points to be saved after a simulation. In the Outputs page you chose from a tree which outputs and parameters to save after a simulation. In the Solver page one can choose which numerical solver to be used to solve the problem as well as to edit its settings. For information about the various solver settings in the Solver page, we refer to the MATLAB documentation.



Pandas - lobj201_Th230

Basic settings

Time unit: year

Start time: -9000.0 year

End time: 60000.0 year

Type of simulation

● Best estimate

○ Probabilistic

○ Simulation table

○ Sensitivity anal...

Number of simulations: 1000

__Figure 5-6.__ Pandas Simulation perspective.

5.6.2 Probabilistic Settings

The probabilistic settings window contains three pages: General, Parameters and Correlation. In the General page, you edit the number of simulations, the seed of the random number generator and the sampling type (Monte Carlo/Latin Hypercube). In the Parameters page, you select the parameters to be included in the analysis; these are uncertain and will be sampled instead of being fixed. Only parameters that have been assigned probability density functions can be selected. In the Correlation page, correlations between parameters can be assigned in order to reduce noise and to avoid impossible parameter combinations. A probabilistic simulation starts with sampling values for all the probabilistic parameters, using either a (random) Monte Carlo scheme or Latin Hypercube sampling. After the values are sorted so that correlated parameter values are used in the same iteration. For example, consider the parameters weight and height. These parameters are log-normally distributed and are positively correlated, meaning that it is common for heavy humans to also be tall. By assigning a correlation of say 0.5, the likelihood of the weight parameter being large in the same iteration as the height parameter is large will increase. To add correlations you must first select the Enabled button. You then click the Add button for each parameter pair, select the two parameters from the left and right lists and enter a correlation between the two.

5.6.3 Simulation table Settings

The simulation table is organized into rows, one for each iteration (or parameter set), and columns, one for each parameter. Parameters can be added by pressing the Add parameter button. Select one or more parameters from the window that appears. When you have many parameters in your model, you can search for parameters in the field below. Iterations are added by pressing the Add iteration button.

5.6.4 Sensitivity Analysis Settings

The sensitivity analysis settings window contains two pages: Method and Parameters. The Parameters page is the same as in the probabilistic settings window. In the Method window the type of sensitivity analysis method to be performed are to be chosen. Depending on the chosen method, various settings for the defined method can be edited.

5.7 Result charts and Result tables

In the Result chart and table perspectives, all simulation results are listed after a simulation has completed and can be analysed graphically by different types of charts and numerically by different types of tables. When you select any outputs, a chart/table called “Quick view” is automatically created. The main use of “Quick view” is to quickly browse simulation outputs. The type of chart or table is determined from the type of output. For instance, if you select a Parameter output an index table is created but if you select a time dependent variable, as a Reservoir, a time table is created. If you want to keep the information in the chart/table, you can just give it another name.

You can create four different types of charts from the Charts perspective:

1. Time chart.
2. Scatter chart.
3. Histogram chart.
4. Tornado chart.

A Time chart plots simulation outputs against time. Outputs that are not time-dependent, such as parameters, will be plotted as a straight line. The other charts listed are only available after a non-deterministic simulation where you have probabilistic data. A Scatter chart plots one or more simulation outputs against a specific output at a specific time point. A Histogram chart displays one or more simulation outputs in the form of a histogram (frequency diagram). A Tornado chart ranks the importance of simulation inputs (parameters) to a specified output (at a specified time point). Figure 5-7 illustrates an example how the Result Chart perspective can look like after a deterministic simulation using a Time chart.

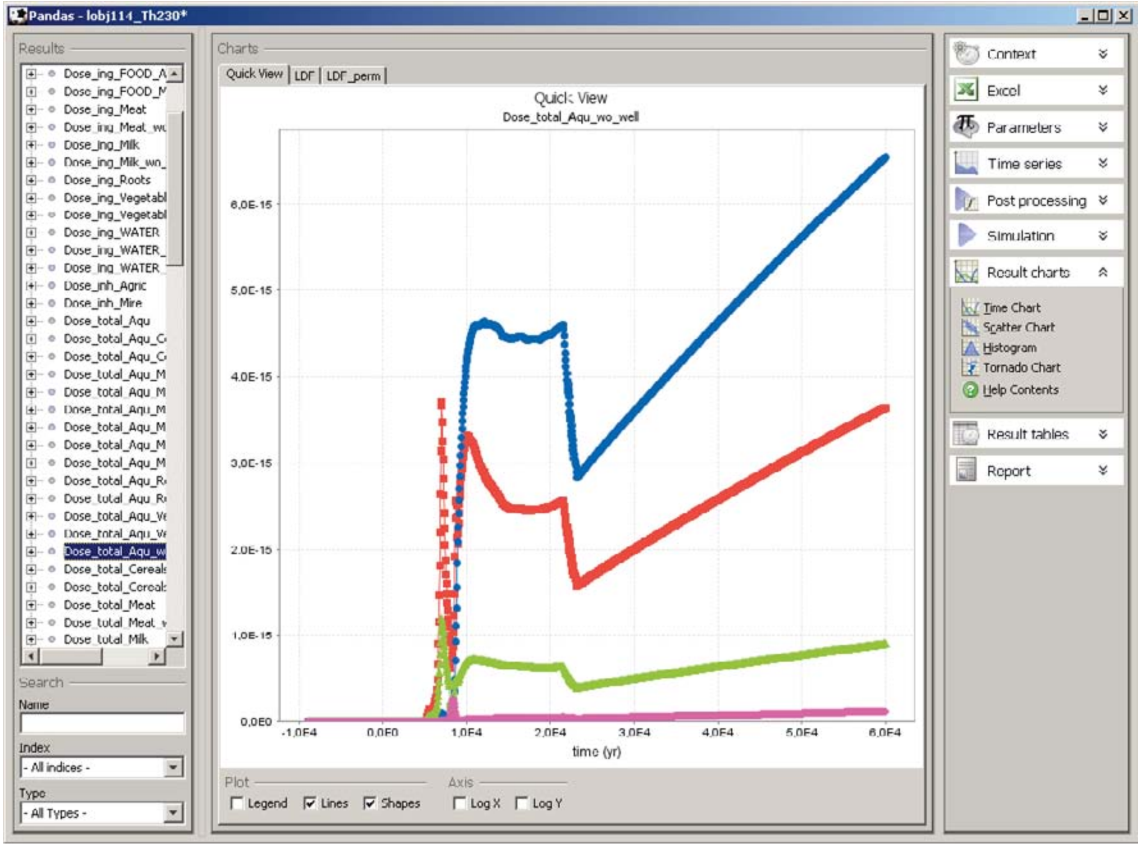


Figure 5-7. Pandas Result charts perspective.

You can create five types of tables from the Tables perspective:

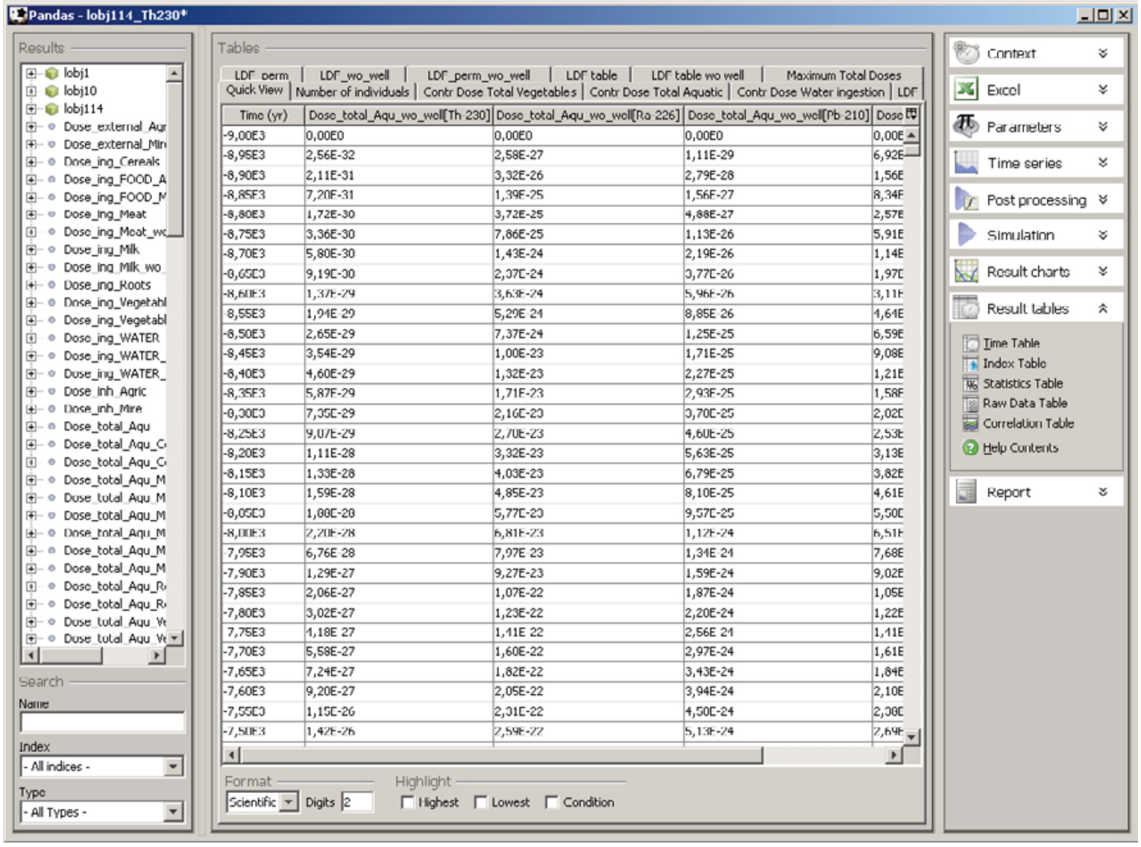
1. Time table.
2. Index table.
3. Statistics table.
4. Correlation table.
5. Raw data table.

In a Time table, time-dependent simulation output data is presented. In an Index table, simulation output data for a specific time-point is presented by index. In a Statistics table, probabilistic simulation output data, for a specific time-point, is presented. A Correlation table displays rank coefficients for all simulation inputs against a selected number of outputs (at a specific time point). A Raw data table presents all probabilistic data for a selected number of outputs. Its main use is when you need to post-process probabilistic data in another tool (such as Excel). Figure 5-8 illustrates an example how the Result Tables perspective can look like after a simulation.

Charts and tables can be created, named and saved and the data will be updated automatically after a new simulation. Charts can also be exported to graphic files whilst tables can be exported to Excel format to be stored locally on computer.

5.8 Report

In the Report perspective, see Figure 5-9, assessment settings can be reviewed in a HTML-format. Listings of all Reservoir, Transfer and Function blocks with settings are also printed. The report can be saved to a HTML-file.

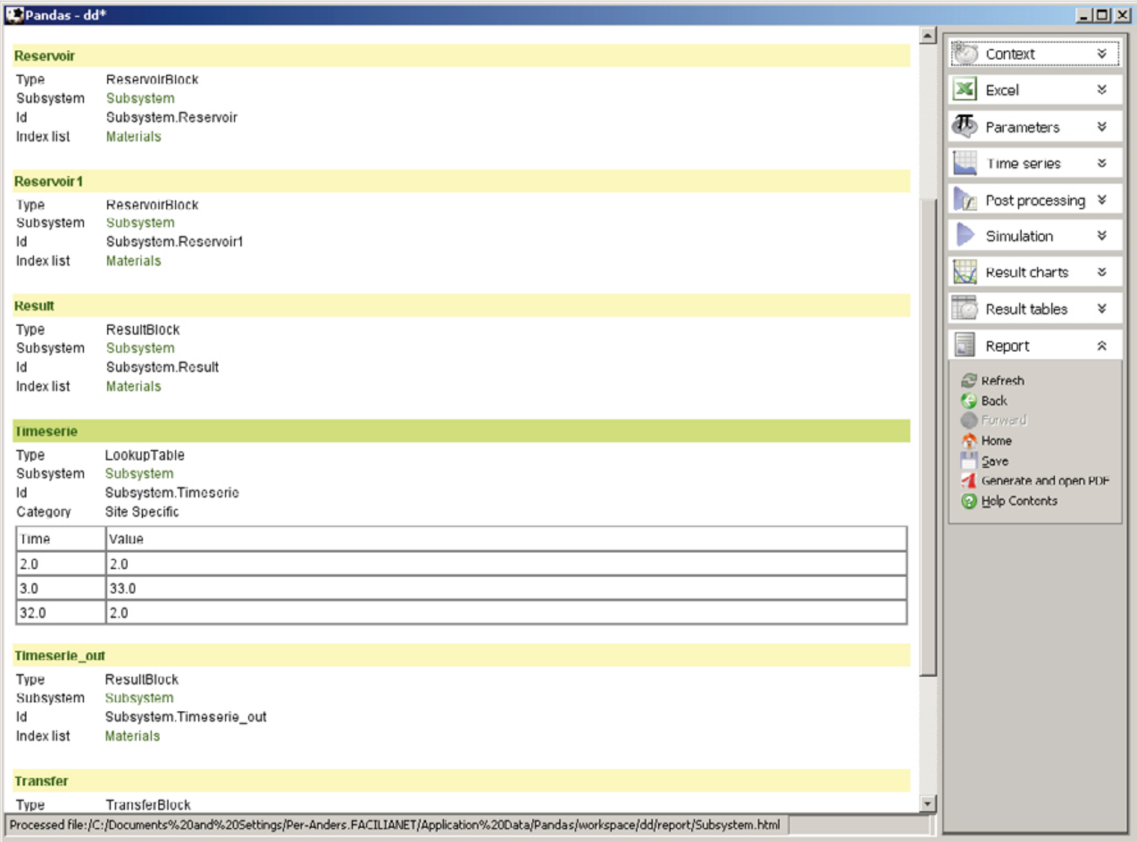


Time (yr)	Dose_total_Aqu_wo_well[Th-230]	Dose_total_Aqu_wo_well[Ra-226]	Dose_total_Aqu_wo_well[Pb-210]	Dose
-9,00E3	0,00E0	0,00E0	0,00E0	0,00E
-8,95E3	2,56E-32	2,58E-27	1,11E-29	6,92E
-8,90E3	2,11E-31	3,32E-26	2,79E-28	1,56E
-8,85E3	7,20E-31	1,39E-25	1,56E-27	8,34E
-8,80E3	1,72E-30	3,72E-25	4,88E-27	2,57E
-8,75E3	3,36E-30	7,86E-25	1,13E-26	5,91E
-8,70E3	5,80E-30	1,43E-24	2,19E-26	1,14E
-8,65E3	9,19E-30	2,37E-24	3,77E-26	1,97E
-8,60E3	1,37E-29	3,63E-24	5,96E-26	3,11E
-8,55E3	1,94E-29	5,29E-24	8,85E-26	4,64E
-8,50E3	2,65E-29	7,37E-24	1,25E-25	6,59E
-8,45E3	3,54E-29	1,00E-23	1,71E-25	9,08E
-8,40E3	4,60E-29	1,32E-23	2,27E-25	1,21E
-8,35E3	5,87E-29	1,71E-23	2,93E-25	1,58E
-8,30E3	7,35E-29	2,16E-23	3,70E-25	2,02E
-8,25E3	9,07E-29	2,70E-23	4,60E-25	2,53E
-8,20E3	1,11E-28	3,32E-23	5,63E-25	3,13E
-8,15E3	1,33E-28	4,03E-23	6,79E-25	3,82E
-8,10E3	1,59E-28	4,85E-23	8,10E-25	4,61E
-8,05E3	1,88E-28	5,77E-23	9,57E-25	5,50E
-8,00E3	2,20E-28	6,81E-23	1,12E-24	6,51E
-7,95E3	6,76E-28	7,97E-23	1,34E-24	7,68E
-7,90E3	1,29E-27	9,27E-23	1,59E-24	9,02E
-7,85E3	2,06E-27	1,07E-22	1,87E-24	1,05E
-7,80E3	3,02E-27	1,23E-22	2,20E-24	1,22E
-7,75E3	4,18E-27	1,41E-22	2,56E-24	1,41E
-7,70E3	5,58E-27	1,60E-22	2,97E-24	1,61E
-7,65E3	7,24E-27	1,82E-22	3,43E-24	1,84E
-7,60E3	9,20E-27	2,05E-22	3,94E-24	2,10E
-7,55E3	1,15E-26	2,31E-22	4,50E-24	2,38E
-7,50E3	1,42E-26	2,59E-22	5,13E-24	2,69E

Format: Scientific, Digits 2

Highlight: □ Highest □ Lowest □ Condition

Figure 5-8. *Pandas Result tables perspective.*



Reservoir

Type ReservoirBlock
Subsystem Subsystem
Id Subsystem.Reservoir
Index list Materials

Reservoir1

Type ReservoirBlock
Subsystem Subsystem
Id Subsystem.Reservoir1
Index list Materials

Result

Type ResultBlock
Subsystem Subsystem
Id Subsystem.Result
Index list Materials

Timeserie

Type LookupTable
Subsystem Subsystem
Id Subsystem.Timeserie
Category Site Specific

Time	Value
2.0	2.0
3.0	33.0
32.0	2.0

Timeserie_out

Type ResultBlock
Subsystem Subsystem
Id Subsystem.Timeserie_out
Index list Materials

Transfer

Type TransferBlock

Processed file:/C:/Documents%20and%20Settings/Per-Anders.FACILIANET/Application%20Data/Pandas/workspace/dd/report/Subsystem.html

Figure 5-9. *Pandas Report perspective.*

6 Building models

Procedure of how one may create models from a model description using Pandora is described in the following sub-sections. The chosen models are a peat-bog model and a coast system. These models are typical basic ecosystem building blocks used to assess the transport and accumulation of radionuclides in the biosphere using Pandora. Final implementations of these two models can be found in the Pandora library.

6.1 Peat-bog model

In this section, we illustrate in high detail how to implement a very simple ecosystem model in Pandora. The chosen model is the Peat bog model also described in /Jones et al. 2004/.

6.1.1 Model specification

The peat module defined in /Bergström et al. 1999/ consists of two separate compartments, one for radionuclides sorbed in water, compartment 1, and one compartment for radionuclides in the solid (particulate) phase, compartment 2. The nuclides enter and leave the system through the water compartment. Conceptual model of the peat module is illustrated in Figure 6-1.

The equation system given in /Bergström et al. 1999/ for the radionuclide transport in the peat module can be written

$dy_1/dt = yin+TC_{12}y_2–TC_{01}y_1–TC_{21}y_1$

$dy_2/dt = TC_{21}y_1–TC_{12}y_2$

With y_1 representing the activity in the compartment with radionuclides sorbed in the void water and y_2 corresponds to the activity of the particulate nuclides. The transfer constant TC_{01} corresponds to the nuclides leaving the system through the runoff water and TC_{21} and TC_{12} to nuclide transport from the desorbed to the particulate phase and in the reverse direction. A detailed description of these transfer coefficients is given in Table 6-2 and all associated input parameters are listed in Table 6-1. The activity concentration in each compartment is given in Table 6-3.

The peat module is defined by a set of seven different properties, see Table 6-1.

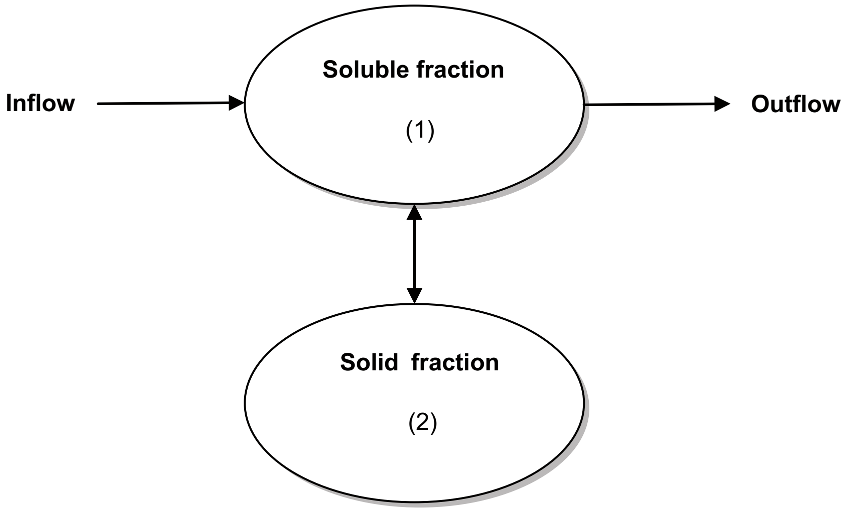


Figure 6-1. *Compartment systems for the peat bog module.*

Table 6-1. Input parameters for the peat module.

Parameter	Variable name in Pandora	Value		Unit
Depth of the peat bog	DEPTH	0.5		m
Area of the peat bog	AREA	10,000		m^2
Density of peat	DENSITY	100		kg/m^3
Porosity of peat	POROSITY	0.9		1
Runoff	RUNOFF	0.24		$m^3/(m^2y)$
Reaction half-time	TK	0.001		y
Distribution factor (concentration of radionuclides on solids compared to dissolved)	KD	Cl-36	0.001	m^3/kgd.w.
		Mo-93	0.1	
		Np-237	0.1	
		I-129	0.3	
		Ni-59	0.5	
		Cs-135	1	
		Pu-239	50	

Table 6-2. Transfer coefficients occurring in the peat module.

Name	Description	Equation
TC_{01}	Transfer coefficient representing horizontal outflow from the soluble compartment.	$R/(e_pD_p)$
TC_{21}	Transfer coefficient representing transport to solid from soluble compartment.	$K_d ln(2)/T_k\ rho_p/e_p$
TC_{12}	Transfer coefficient representing transportation to soluble from solid compartment.	$ln(2)/T_k$
where		**Unit**
R	Runoff	$m^3/(m^2y)$
e_p	Porosity in peat	m^3/m^3
D_p	Depth of peat	m
Kd	Distribution coefficient for radionuclides	m^3/kgd.w.
Tk	Reaction half-time	y
rho_p	Density of peat	kg/m^3

Table 6-3. Activity concentration C for a selected nuclide in compartment of interest for dose calculations in the peat module.

Name	Description	
$C_{Soluble}$	level / e_p * D_p * A	
C_{Solid}	level / rho_p * D_p * A	
where		Unit
level	Amount of selected nuclide in respective compartment	mol
e_p	Porosity in peat	m^3/m^3
D_p	Depth of peat	m
A	Area of peat bog	m^2
rho_p	Density of peat	kg/m^3

6.1.2 Model implementation

A model in Pandora can be implemented in numerous ways. This is one way how to implement a Pandora model from a detailed model specification.

Preparation

First one need to make sure that the Pandora block library and all supporting functions is in the MATLAB search path. If Pandora is not in the MATLAB Path; MATLAB does not know about all the functionalities Pandora offer and it is impossible to create a Pandora model.

The MATLAB Path can be configured from an interface opened from File|Set Path…

Click on the button ‘Add with Subfolders…’ and choose the folder where Pandora is located on your computer.

Click on Save and Close.

Starting up

Start The Pandora Manager by writing pandora2 and hitting Enter in the MATLAB command prompt.

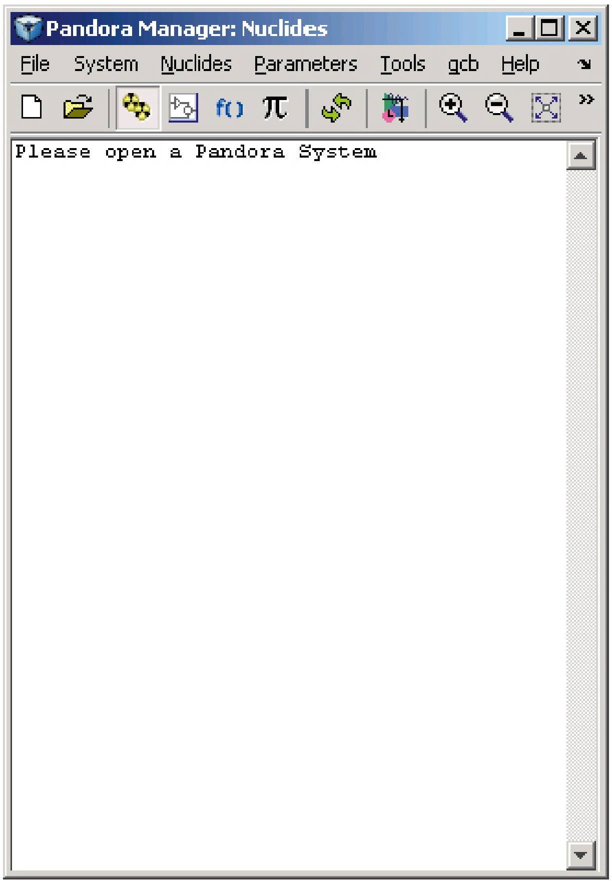
```
>> pandora2
```

The Pandora Manager then opens as illustrated in Figure 6-2.

To open up the Simulink Block Library one can either click on the Simulink symbol or go to Tools|Simulink Browser. After doing this the Simulink Browser opens as illustrated in Figure 6-3.

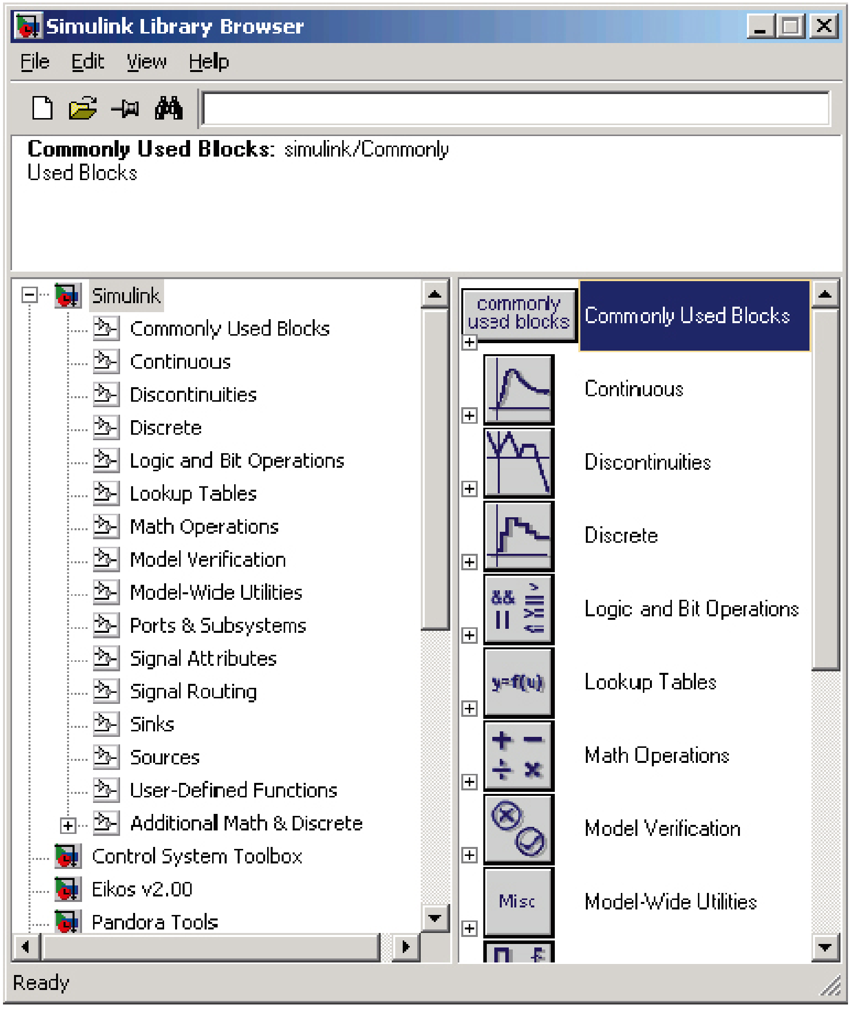
After clicking on Pandora Tools in the hierarchy tree to the left in the Simulink Browser, you will have access to all Pandora Blocks in the right part of the Simulink Browser.

Now you will have the Pandora Manager and the Simulink Library Browser open. The next step is to create a new model. There are several ways in doing this. From the Pandora Manager one can either click on the New Model sign in the Toolbar menu or choose File|New Model in the standard menu. One can actually do the same two steps from the Simulink Browser. If doing it from the Simulink Browser the model is a standard Simulink model. To make it a Pandora model one need to drag and drop a Pandora Block from the Simulink Library Browser into the model. To get a connection to the model from the Pandora Manager you go to System in the standard menu and choose the newly created model.



```
Please open a Pandora System
```

Figure 6-2. *Pandora Manager Interface.*



Commonly Used Blocks: simulink/Commonly Used Blocks

Figure 6-3. Simulink Browser.

Adding species of interest

In the Pandora block in the newly created model there are two radionuclides already defined, U-234 and Th-230. These are not the ones of interest in this exercise. We want to model Cl-36, Mo-93, Np-237, I-129, Ni-59, Cs-135 and Pu-239. To change this we could either do it manually by double-clicking on the Pandora block and adjusting the fields therein. However, it is much easier to use the Pandora Manager in doing this.

So from the Pandora Manager choose Nuclides|Select Nuclides in the standard menu. In the newly opened interface move the two nuclides chosen in the right part to the left using the arrows button <<. And move the seven nuclides of interest to the right using the other arrows button >> and click OK.

Adding reservoir blocks

Now we want to add two reservoir blocks in the model. These will represent y_1 and y_2 in the ordinary differential equation mentioned above.

Drag and drop a Reservoir Block from the Simulink Library Browser into your model. Right click-hold and drag a copy of it somewhat below and release the button. While still having the same block marked hold down Ctrl and hit R two times. This will rotate the block 180 degrees.

Double click on the top reservoir so that the Mask Dialog appears. Write Soluble_Fraction in the Name field, click OK. Do the same thing for the other reservoir and call it Solid_Fraction. Now your model should look similar to Figure 6-4.

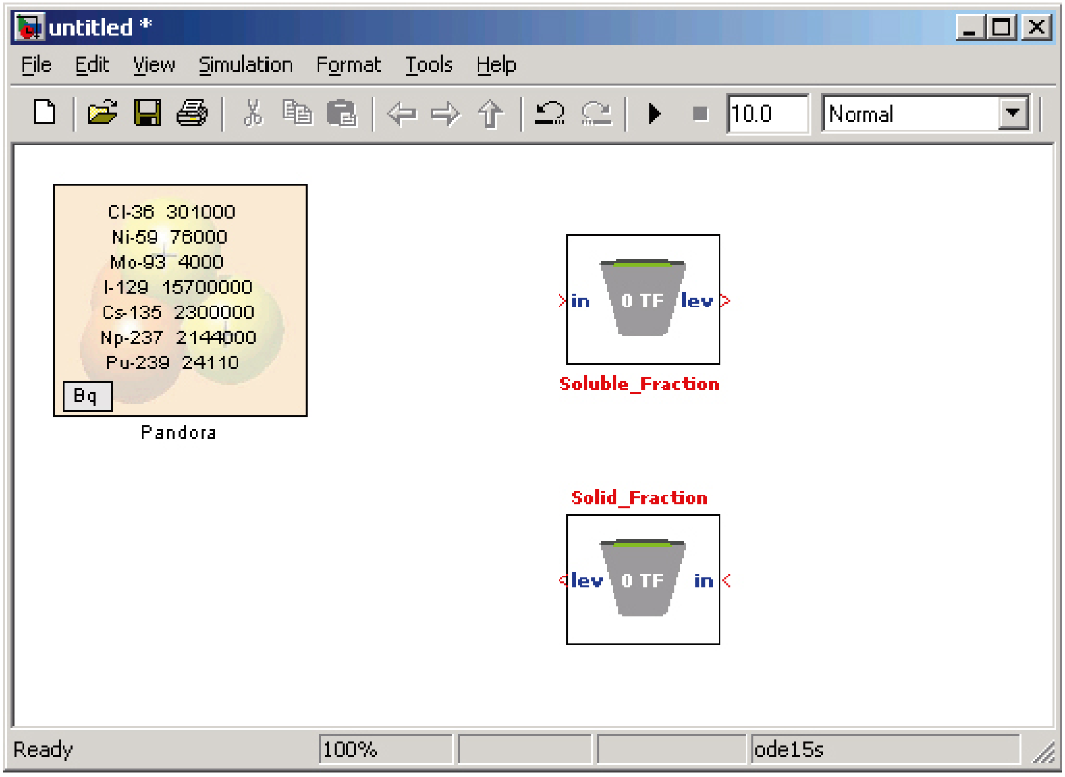


Figure 6-4. *Model with two reservoirs.*

Adding transfer blocks

A flux in a model represents a change in the differential equation. Therefore, for each time-step the fluxes are the right hand sides of the differential equations. Changes coming from the radioactive decay of the radionuclides are computed inside Pandora. Fluxes out from a reservoir Block are modelled using transfer blocks.

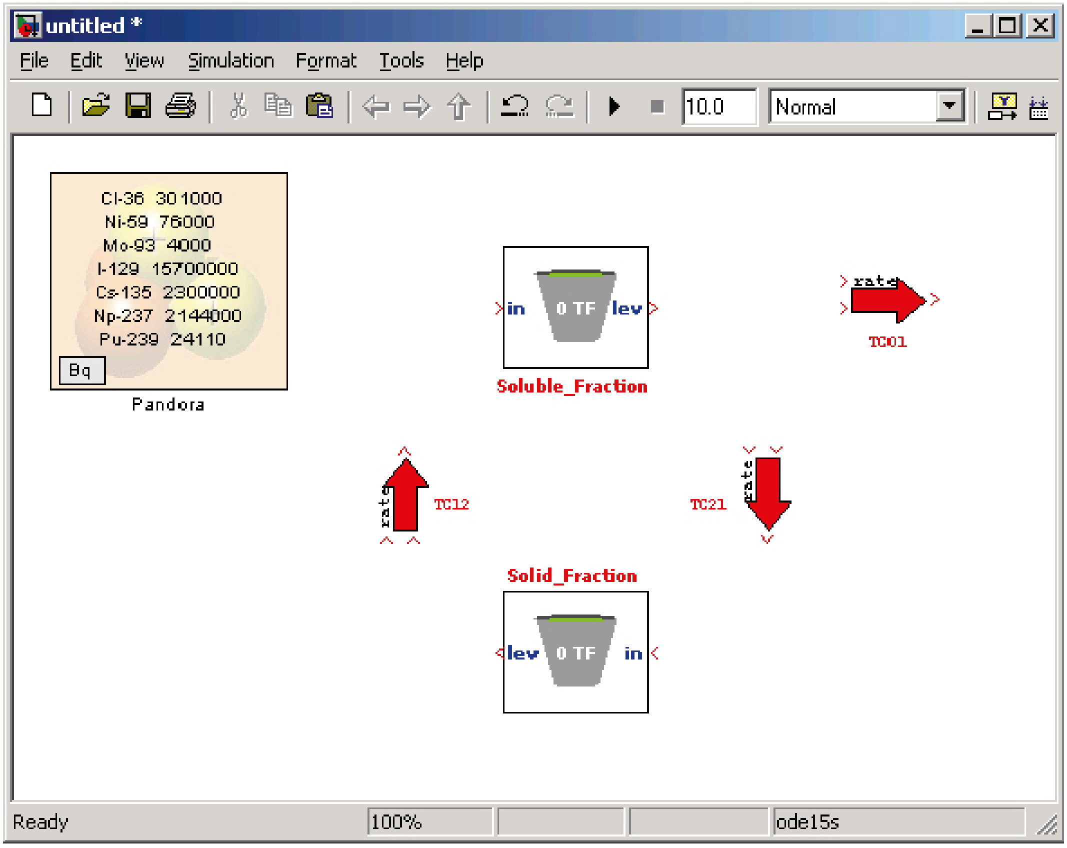
In our Peat-bog model we have three fluxes; TC_{01}, TC_{21} and TC_{12} representing the flux out from the soluble fraction and the fluxes in between the soluble and the solid fraction.

We begin by adding three Transfer blocks in good positions to the model and naming them TC01, TC21 and TC12. Now the model will look similar to Figure 6-5.

Connect lines between the out ports of the Reservoir blocks with the second in ports of the Transfer Blocks. Now the model will look similar to Figure 6-6. The blocks are not immediately updated when just connecting lines to them. Blocks are updated at model simulation, when their mask dialog is opened or if you choose to manually Update Diagram (Ctrl-D) from the Edit menu in Model standard menu.

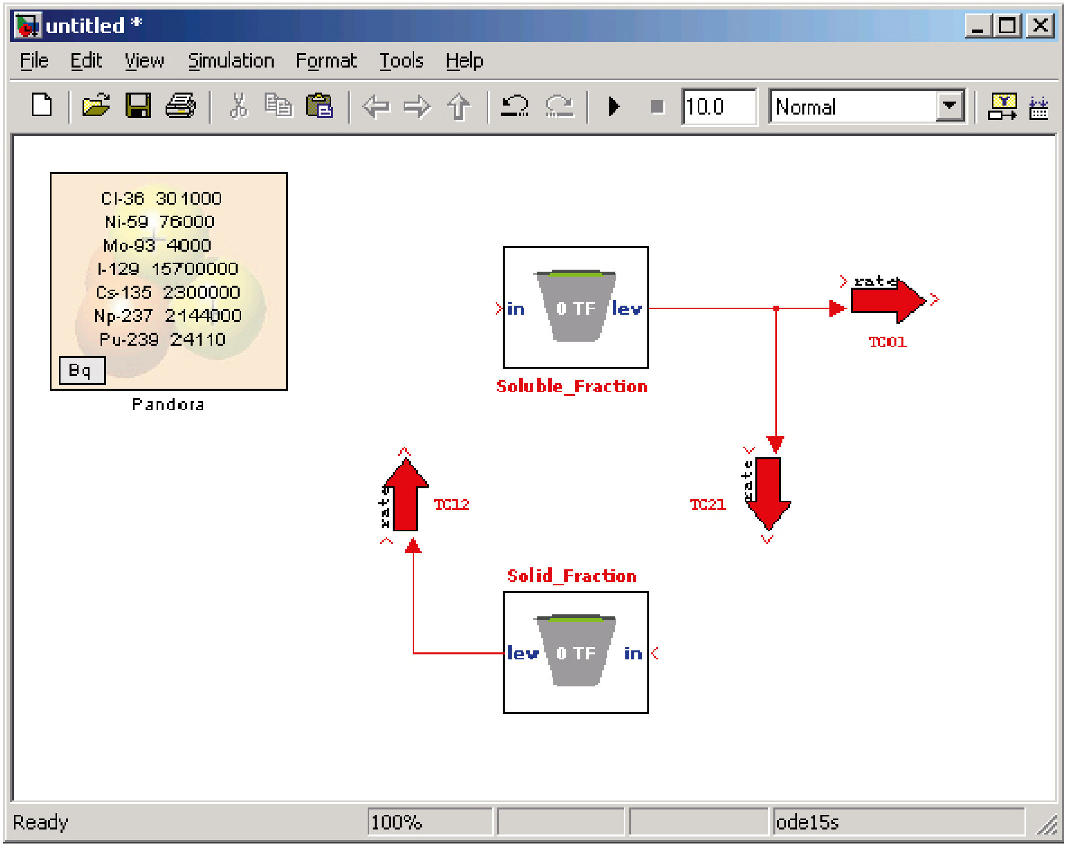
Click Ctrl-D and the model will be updated and look similar to Figure 6-7.

As can be seen from Figure 6-6 and Figure 6-7, the colours of the Transfer blocks changed from red to gray and the numbers on the buckets in the Reservoir blocks updated to represent the number of outgoing Transfer blocks that are connected to them.



Cl-36	301000
Ni-59	76000
Mo-93	4000
I-129	15700000
Cs-135	2300000
Np-237	2144000
Pu-239	24110

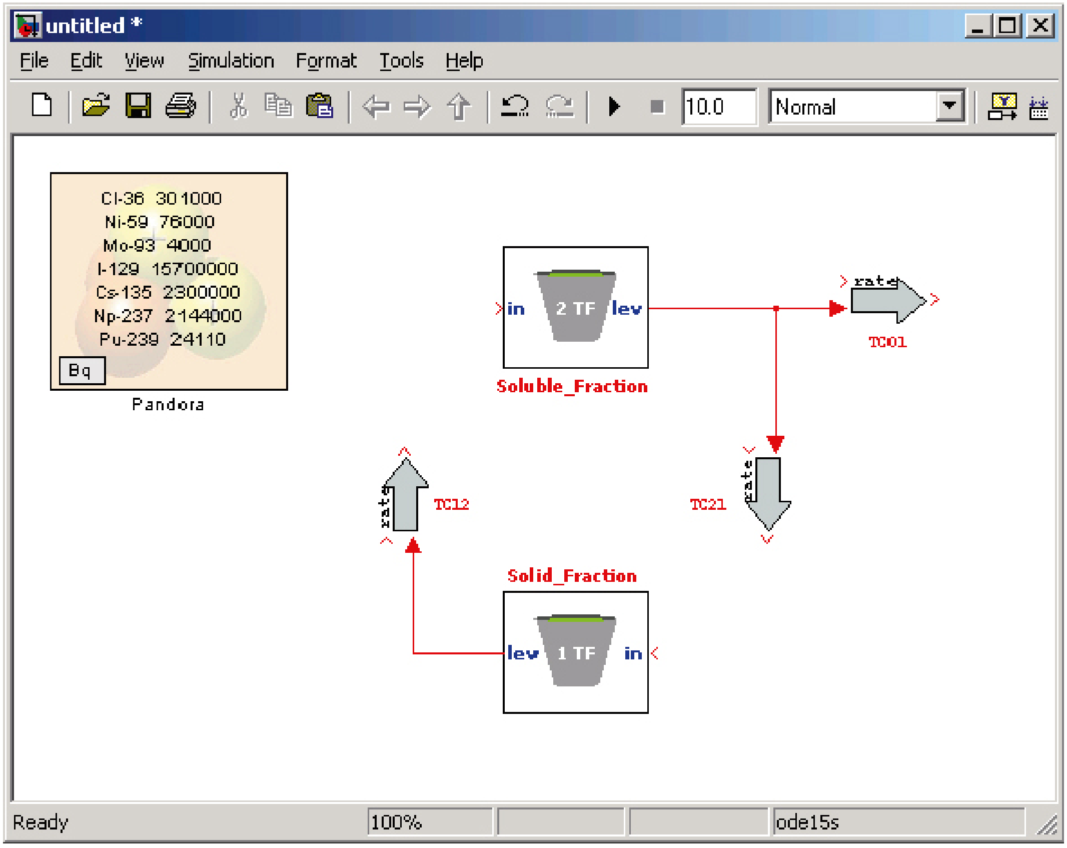
Figure 6-5. Model with two reservoirs and three unconnected fluxes.



Cl-36 301000
Ni-59 76000
Mo-93 4000
I-129 15700000
Cs-135 2300000
Np-237 2144000
Pu-239 24110

10.0 Normal ode15s

Figure 6-6. *Model with two reservoirs and three connected fluxes.*



Cl-36 301000
Ni-59 76000
Mo-93 4000
I-129 15700000
Cs-135 2300000
Np-237 2144000
Pu-239 24110

10.0 Normal ode15s

Figure 6-7. *Updated model with two reservoirs and three connected fluxes.*

Adding function blocks as transfer coefficients

The amount of material transferred through the Transfer block is defined using the other in port in the block. Since the transfer coefficients are equations we choose to use the Function Block for this. Therefore, we add one Function block for each Transfer block and connect them to their respective port named rate. For each Function block double-click on it to open their Mask Dialog and add a reasonable name as well as their equation. Equations used are: RUNOFF/(POROSITY*DEPTH) for function representing rate for TC01, KD*log(2)/TK*DENSITY/POROSITY for function representing rate for TC21 and log(2)/TK for function representing rate for TC12. As you can see the function block automatically adjusts its in ports with the missing variables.

Your model should look similar to Figure 6-8.

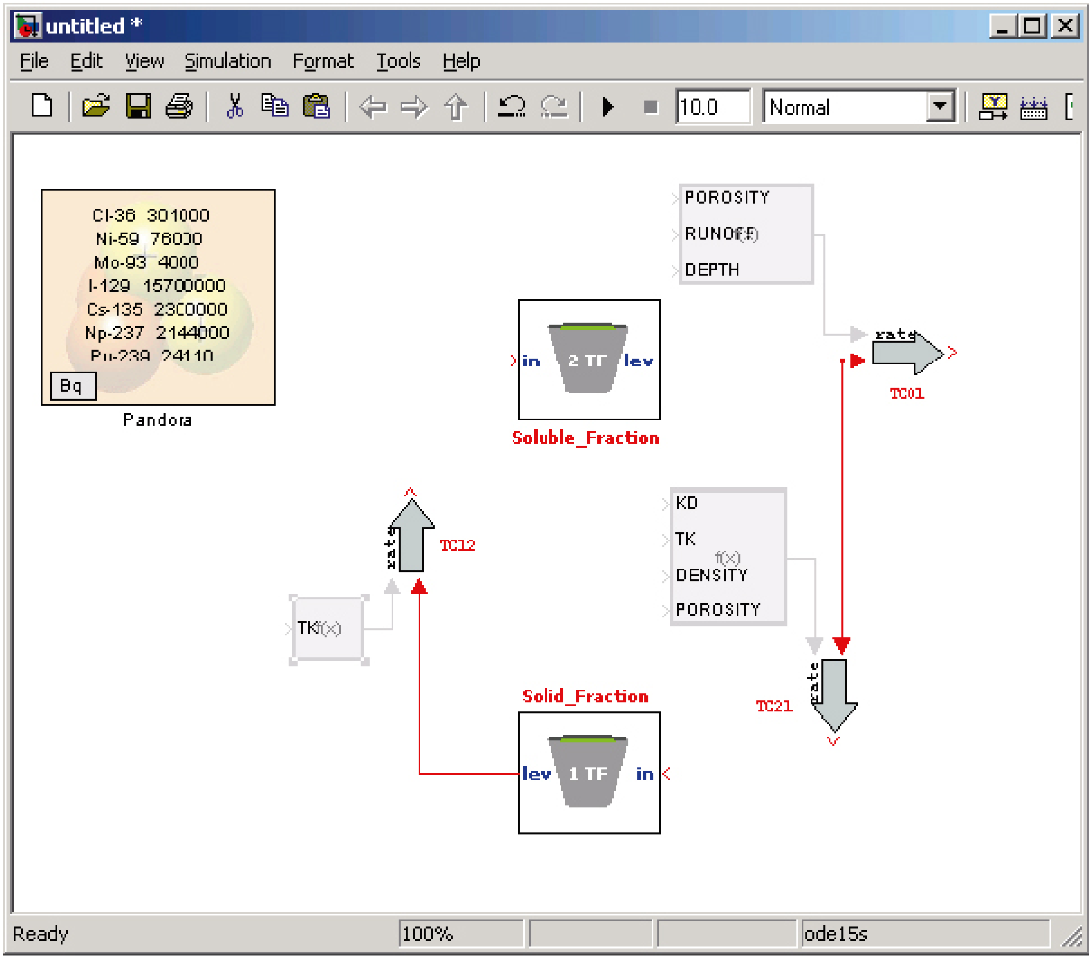


Figure 6-8. *Peat model with added Function Blocks.*

Connecting influxes to the Reservoirs

According to the model specification, we will have an inflow into the model going to the soluble fraction. Therefore, we will have two fluxes going into the soluble fraction, one from outside and one coming from the solid fraction. We use a Function block to sum these two fluxes. We add a Function block named inflow_sum with the expression inflow+TC12. This block we connect to the in port named in in the soluble fraction. We also connect the out ports of the Transfer functions to either in port of inflow_sum or in port of Solid_Fraction and your model should now look similar to Figure 6-9.

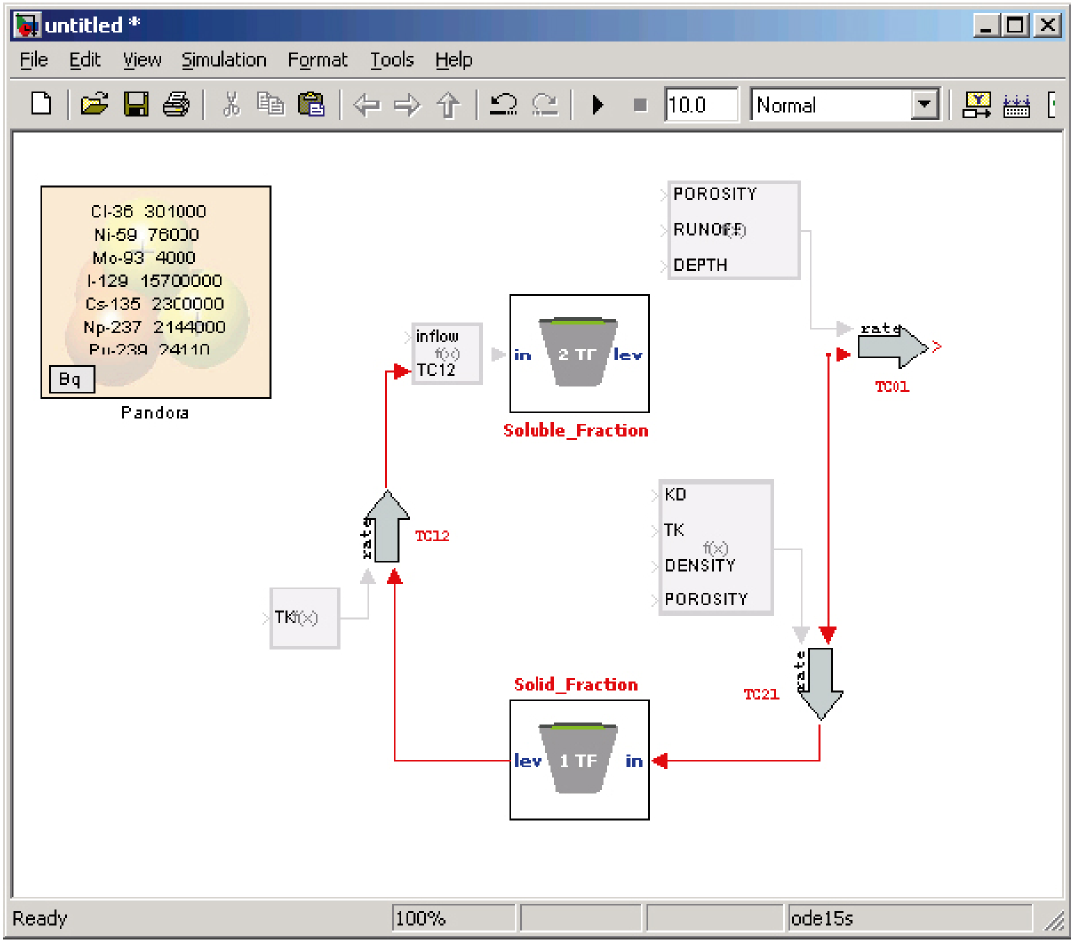
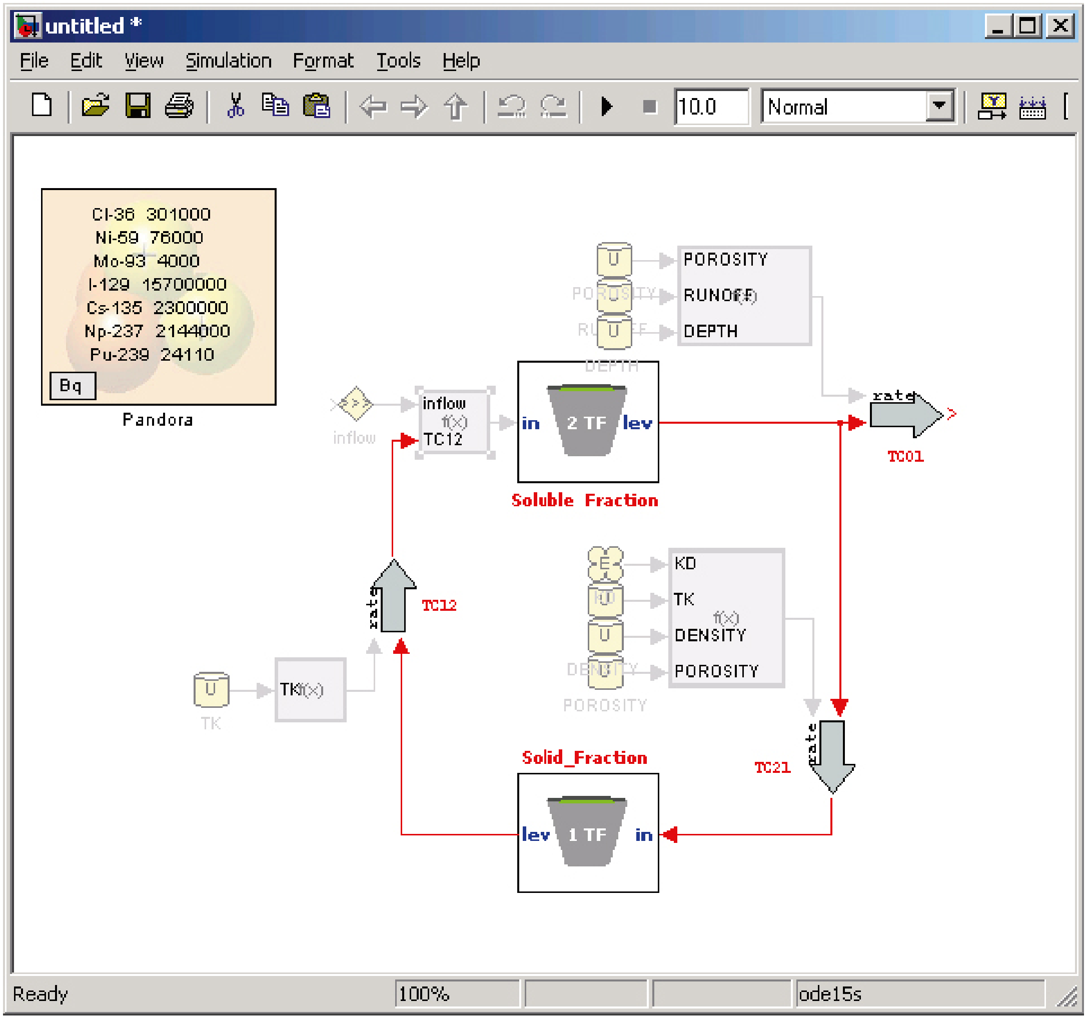


Figure 6-9. *Peat model with all fluxes connected.*

Adding parameters to the model

One can either drag and drop parameters and name and connect them manually in the model. One can also use the GCB menu feature in the Pandora Manager. We choose to use Universal Constants for all parameters except for KD, which is an Element Specific parameter. For the inflow, we choose to add an Interface block. To use the GCB menu you first need to mark a function block, and then click in the menu and choose how and to where to connect blocks.

Now your model should look similar to Figure 6-10.



Cl-36	301000
Ni-59	76000
Mo-93	4000
I-129	15700000
Cs-135	2300000
Np-237	2144000
Pu-239	24110

Figure 6-10. *Peat model with all parameters connected.*

Make a subsystem of the model

Now we are almost finished building the model. To be able to work easier with this ecosystem together with other systems and maybe to save it as a library model we want to put it into a subsystem. To do this we want to mark everything except the Pandora block (Ctrl-A, middle-click on Pandora block). Then we click on a block and choose Create Subsystem from the pop-up menu. Rename your newly created subsystem to peat.

Now your top system model should look like Figure 6-11.

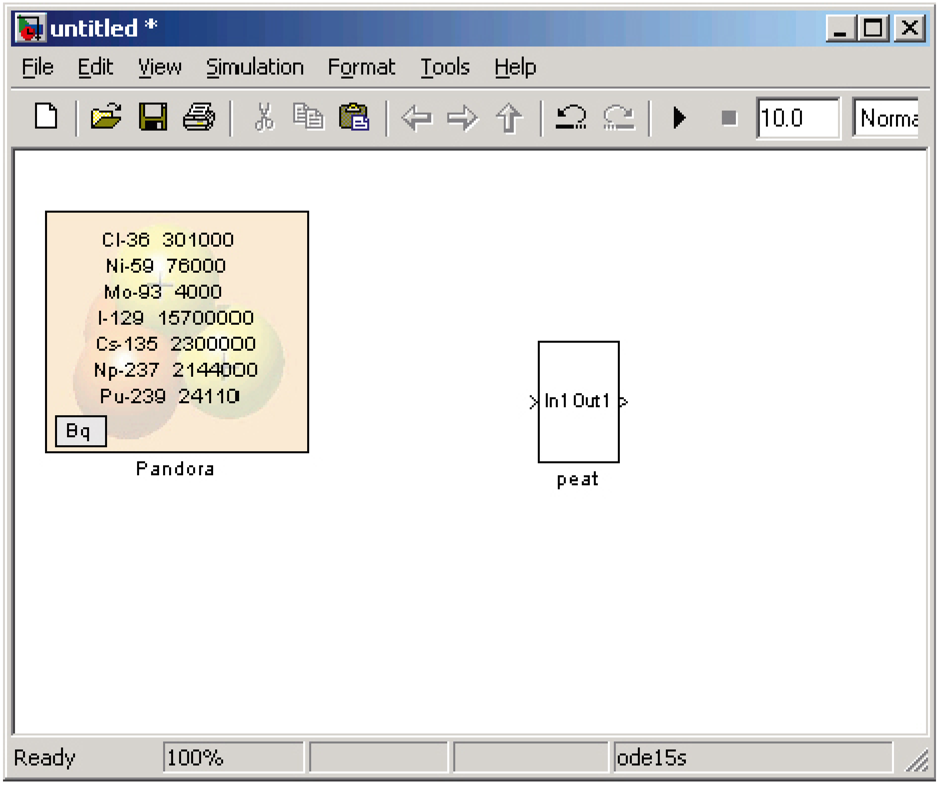


Figure 6-11. *Top level system of Peat model.*

Adding numerical values referenced by parameter variables

To being able to run the model, you need to map numerical values to the parameter variables in the constant+ blocks. One way to do this is to use the Parameter Manager, which is opened from the menu Parameters|Open Parameter Manager in the Pandora Manager. The first time this is done in a MATLAB session all Java libraries needed to use this functionality are read into MATLAB. Therefore, you need to open it again the first time.

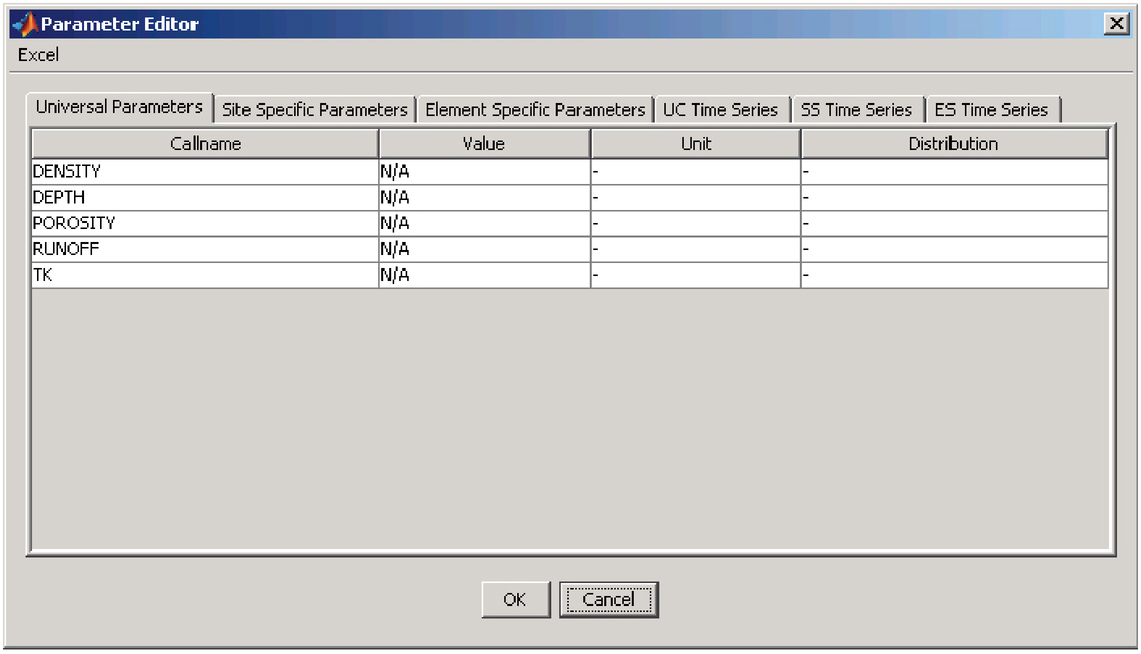
When you get it open, it will look like Figure 6-12.

Fill in correct values for all Universal Parameters and for all the radionuclides in the Element Specific parameter Kd. Then click OK.

Running a simulation

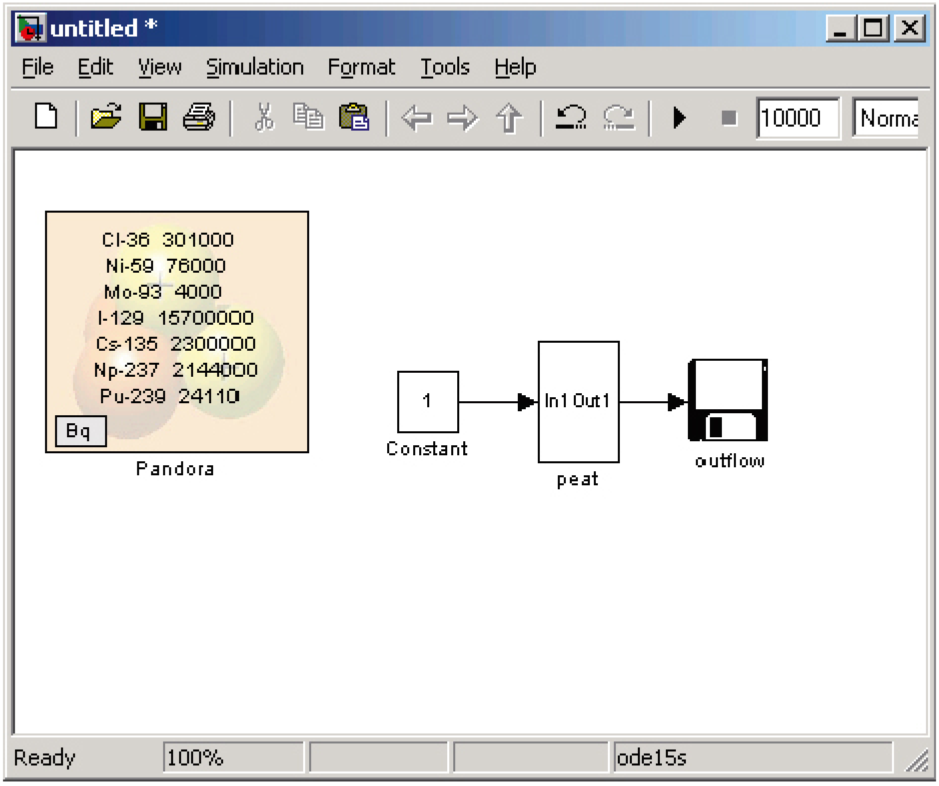
Now the model is ready to run. However, nothing interesting will happen if just clicking on the run button. We have not added any input to the system. To add a constant release rate of 1 Bq/year; add a Constant block from Simulink built-in Library to the top system and connect it to the in port of the peat subsystem. To view how inventory changes over time in the system and how the flux out of the system changes one can go to the Tools menu in Parameter Manager and choose Turn on Data Logging for all Reservoir and also add a Result block connected to the out port of the peat model calling it outflow.

Choose to simulate over 10,000 years and run the simulation. Double-click on a Result block or click on the Result Plot icon in the Toolbar menu in Pandora Manager to open the Result Viewer interface, see Figure 6-14. The simulated data should been automatically loaded into the viewer. If not click on the 'Open…'-button and choose simulation structure and click OK.



Callname	Value	Unit	Distribution
DENSITY	N/A	-	-
DEPTH	N/A	-	-
POROSITY	N/A	-	-
RUNOFF	N/A	-	-
TK	N/A	-	-

Figure 6-12. *Parameter Manager.*



Cl-36 301000
Ni-59 76000
Mo-93 4000
I-129 15700000
Cs-135 2300000
Np-237 2144000
Pu-239 24110

Figure 6-13. Peat model with constant inflow.

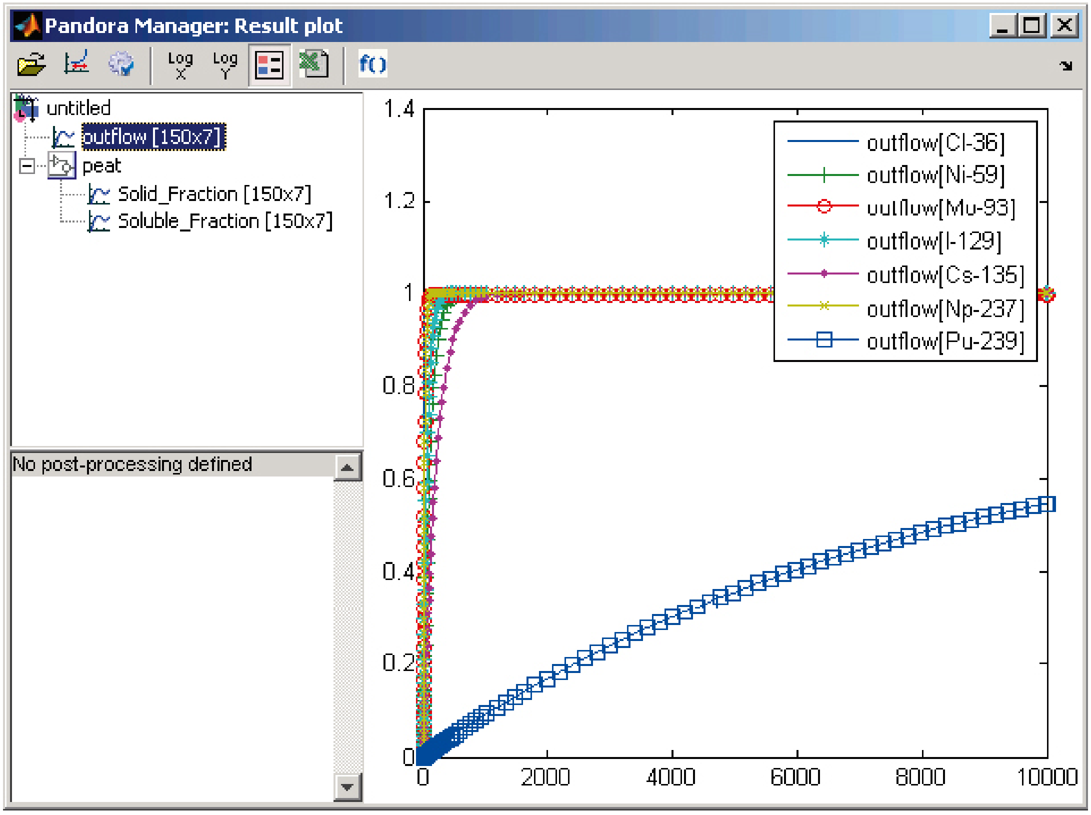


Figure 6-14. Result Viewer with results from Peat model.

6.2 A coast system

This test case model is based on the coast model described in /Karlsson et al. 2000/ and /Bergström et al. 1999/. The coast is modelled with six compartments as shown in the conceptual model in Figure 6-15 and system of equations defined in Equation 6-1. The default parameter values for the test use are given in Table 6-4 and Table 6-5.

6.2.1 Model specification

The system of equations defining the mathematical representation of the test case models is defined in Equation 6-1.

Equation 6-1. ODE of coast system.

$$\begin{cases} \dfrac{d(WaterIC)}{dt} = Inflow + TC_{sw} * SedIC + TC_{oi} * WaterOC - TC_{io} * WaterIC - TC_{ws} * WaterIC \\ \dfrac{d(SedIC)}{dt} = TC_{ws} * WaterIC - TC_{sw} * SedIC - TC_{sd} * SedIC \\ \dfrac{d(DeepSedIC)}{dt} = TC_{sd} * SedIC \\ \dfrac{d(WaterOC)}{dt} = TC_{io} * WaterIC + TC_{sw} * SedIC - TC_{oi} * WaterOC - TC_{ws} * WaterOC \\ \dfrac{d(SedOC)}{dt} = TC_{ws} * WaterOC - TC_{sw} * SedOC \\ \dfrac{d(DeepSedOC)}{dt} = TC_{sd} * SedOC \end{cases}$$

where

$$TC_{io} = \frac{365}{RETTIME}$$

$$TC_{sd} = \frac{SR}{Msed} - TC_{sw}$$

$$TC_{ws} = \frac{K_d \cdot SR}{D \cdot (1 + K_d \cdot Susp)}$$

$$TC_{oi} = \frac{365}{RETTIME} \cdot \frac{V_b}{V_o}$$

$$TC_{sw} = RESUSP$$

$$TC_{out} = OUTFLOW$$

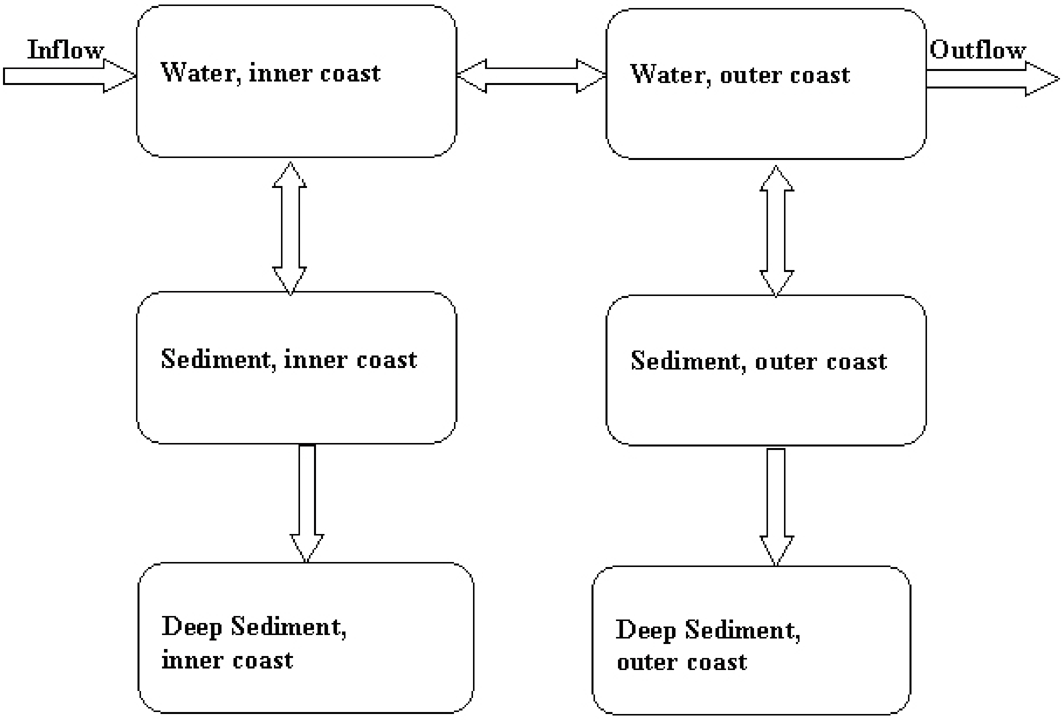


Figure 6-15. Conceptual representation of the coast system model.

Table 6-4. Site-specific parameter default values for the coast system model.

Description	Symbol	Deterministic value	Unit
Area of surface, bay	a	1.4e6	m^2
Mean depth, bay	D	2.3	m
Mean residence time of water in bay	RETTIME	45	days
Volume open seacoast box	V_o	1.7e8	m^3
Mean depth, open sea	D	7	m
Sedimentation rate, bay	SR	2.0	kg/(m^2*year)
Sedimentation rate, sea	SR	0.2	kg/(m^2*year)
Suspended matter, bay	Susp	0.001	kg/m^3
Suspended matter, sea	Susp	0.001	kg/m^3
Mass of top sediment	Msed	10	kg/m^2
Resuspension rate from sediment	RESUSP	0.2	year $^{-1}$
Outflow rate, sea	OUTFLOW	44	year $^{-1}$

Table 6-5. Element-specific parameter default values for Kd of sea sediments (Kd_sea) in the coast system model.

Cl-36	Cs-135	I-129	Mo-93	Ni-59	Np-237	Pu-239
0.001	10	0.3	0.001	10	10	100

6.2.2 Model implementation

In this section, the test model defined in Section 6.2.1 is implemented in Pandora systematically. We assume that the user is familiar with the basics of MATLAB/Simulink. Finally, the implemented model of the coast will correspond to the illustrations in Figure 6-16, Figure 6-17 and Figure 6-18.

The instructions are given on this page and the corresponding figures on the following ones.

1. Start the Pandora Manager by typing pandora2 at the MATLAB command prompt.
2. Create a new Pandora model and save it as coastsystem.mdl.
3. Open the Simulink library browser.
4. Select the Pandora library.
5. Create a subsystem InnerCoast and add blocks and connections according to Figure 6-17.
6. Create a subsystem OuterCoast and add a blocks according to Figure 6-18. Note that you can also copy it partly from the Inner coast subsystem.
7. Connect the two subsystems according to Figure 6-16 and add correct set of radionuclides in the Pandora Block using the Pandora Manager, an Element-specific Constant+ block for inflow and two Plot blocks.
8. Input the parameter values using the Parmeter Manager from the Pandora Manager. Put in element-specific and site-specific parameter values according to Figure 6-19 and Figure 6-20.
9. Set the simulation time-span to be 10,000 years and run the model.
10. When simulation is ready, the two Plot Blocks should give results as in Figure 6-21 and Figure 6-22.

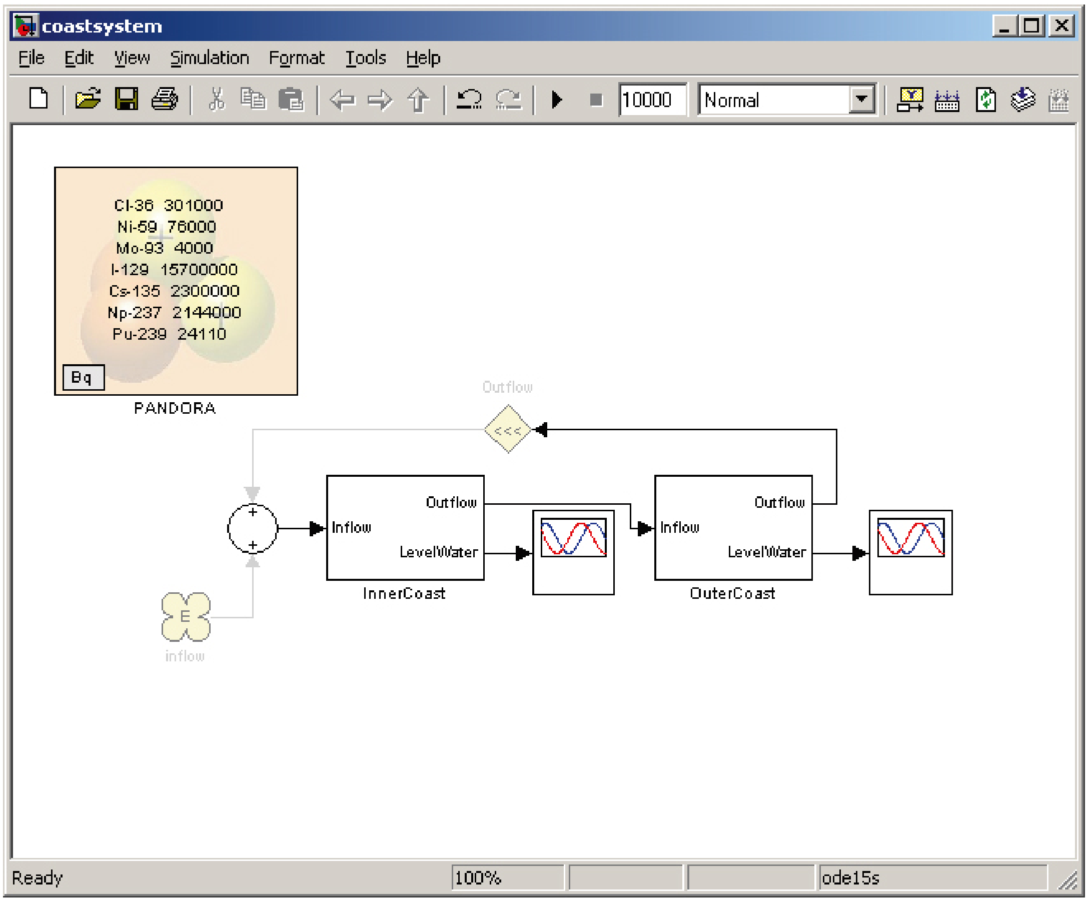


Figure 6-16. *Top level system, connection of the two sub-systems Inner Coast and Outer Coast.*

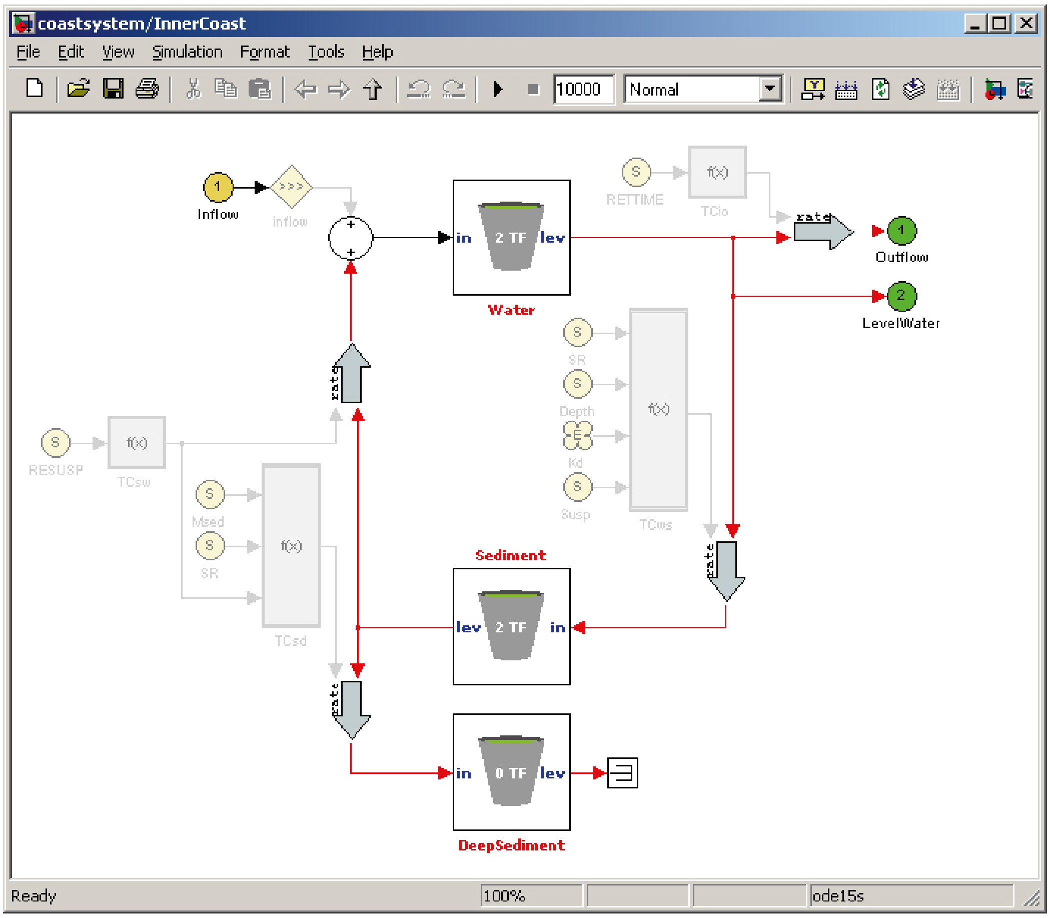


Figure 6-17. *Pandora implementation of the Inner Coast sub-system.*

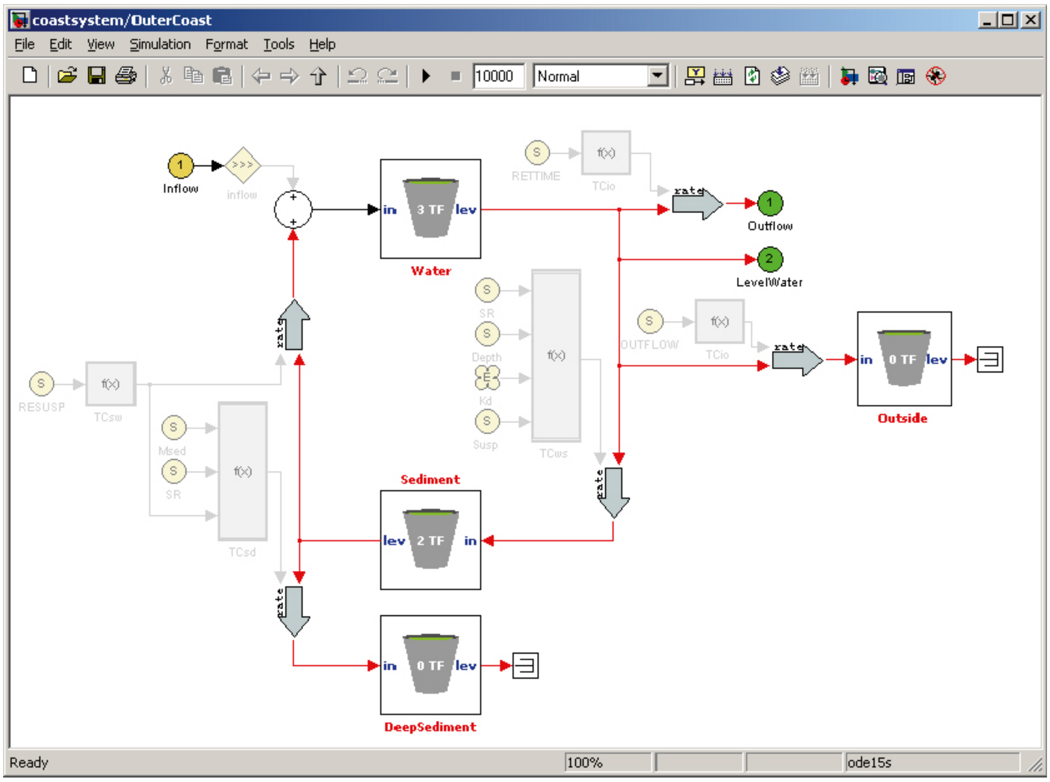
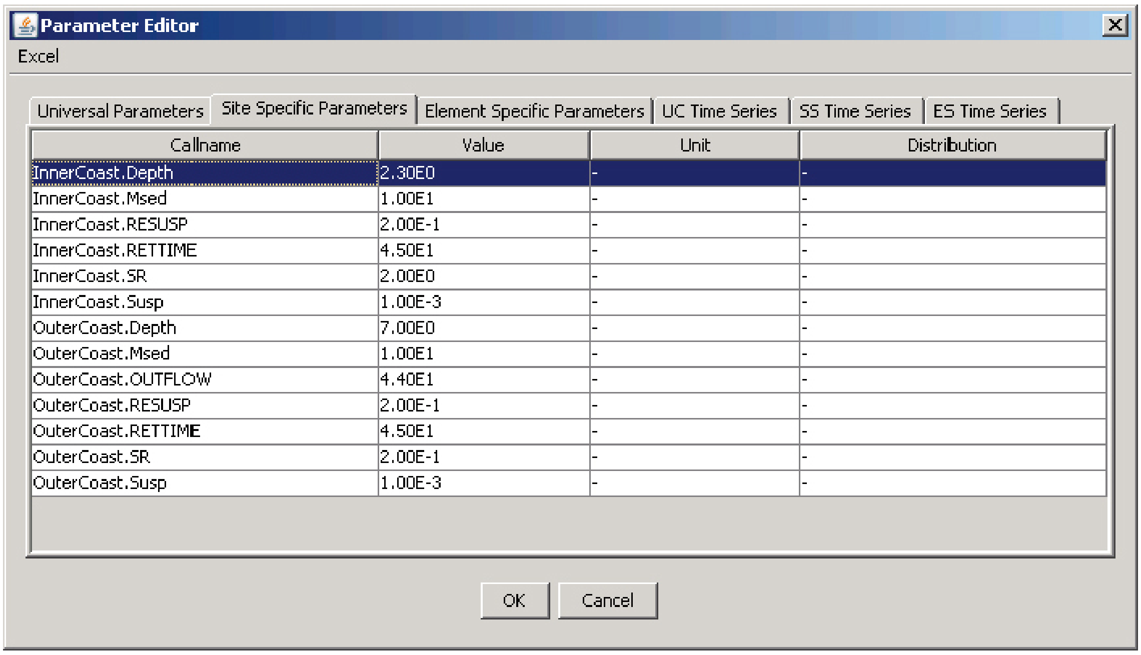
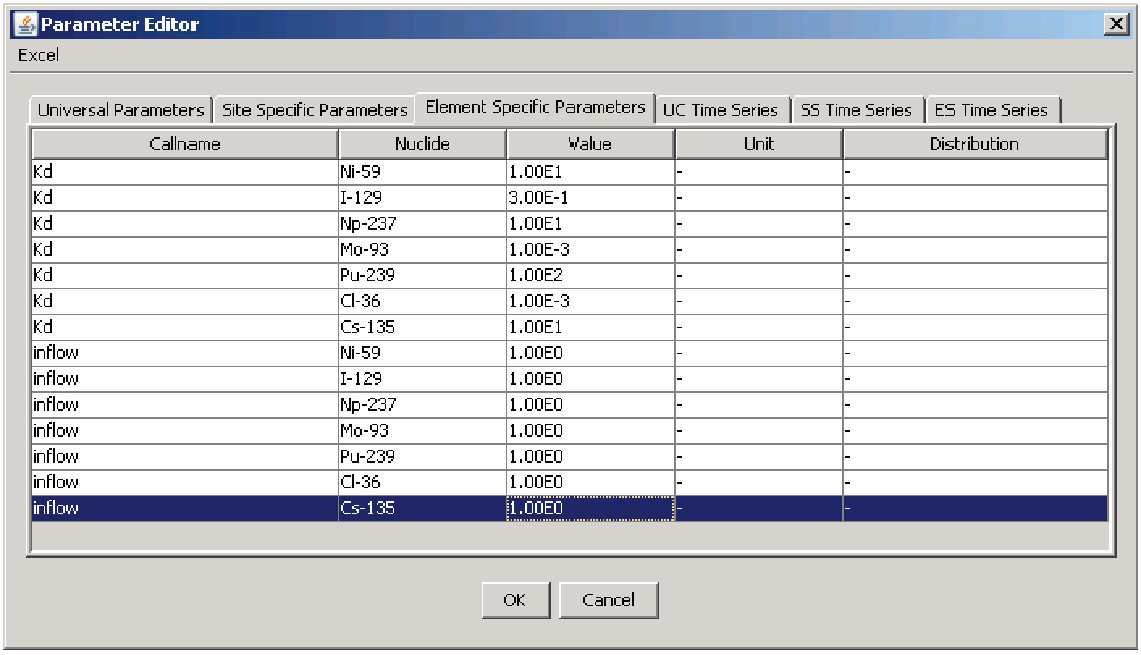


Figure 6-18. *Pandora implementation of the Outer Coast sub-system.*



Callname	Value	Unit	Distribution
InnerCoast.Depth	2.30E0	-	-
InnerCoast.Msed	1.00E1	-	-
InnerCoast.RESUSP	2.00E-1	-	-
InnerCoast.RETTIME	4.50E1	-	-
InnerCoast.SR	2.00E0	-	-
InnerCoast.Susp	1.00E-3	-	-
OuterCoast.Depth	7.00E0	-	-
OuterCoast.Msed	1.00E1	-	-
OuterCoast.OUTFLOW	4.40E1	-	-
OuterCoast.RESUSP	2.00E-1	-	-
OuterCoast.RETTIME	4.50E1	-	-
OuterCoast.SR	2.00E-1	-	-
OuterCoast.Susp	1.00E-3	-	-

Figure 6-19. *Input of Site-specific parameters using the Parameter Manager.*



Callname	Nuclide	Value	Unit	Distribution
Kd	Ni-59	1.00E1	-	-
Kd	I-129	3.00E-1	-	-
Kd	Np-237	1.00E1	-	-
Kd	Mo-93	1.00E-3	-	-
Kd	Pu-239	1.00E2	-	-
Kd	Cl-36	1.00E-3	-	-
Kd	Cs-135	1.00E1	-	-
inflow	Ni-59	1.00E0	-	-
inflow	I-129	1.00E0	-	-
inflow	Np-237	1.00E0	-	-
inflow	Mo-93	1.00E0	-	-
inflow	Pu-239	1.00E0	-	-
inflow	Cl-36	1.00E0	-	-
inflow	Cs-135	1.00E0	-	-

Figure 6-20. *Input of Element-specific parameters using the Parameter Manager.*

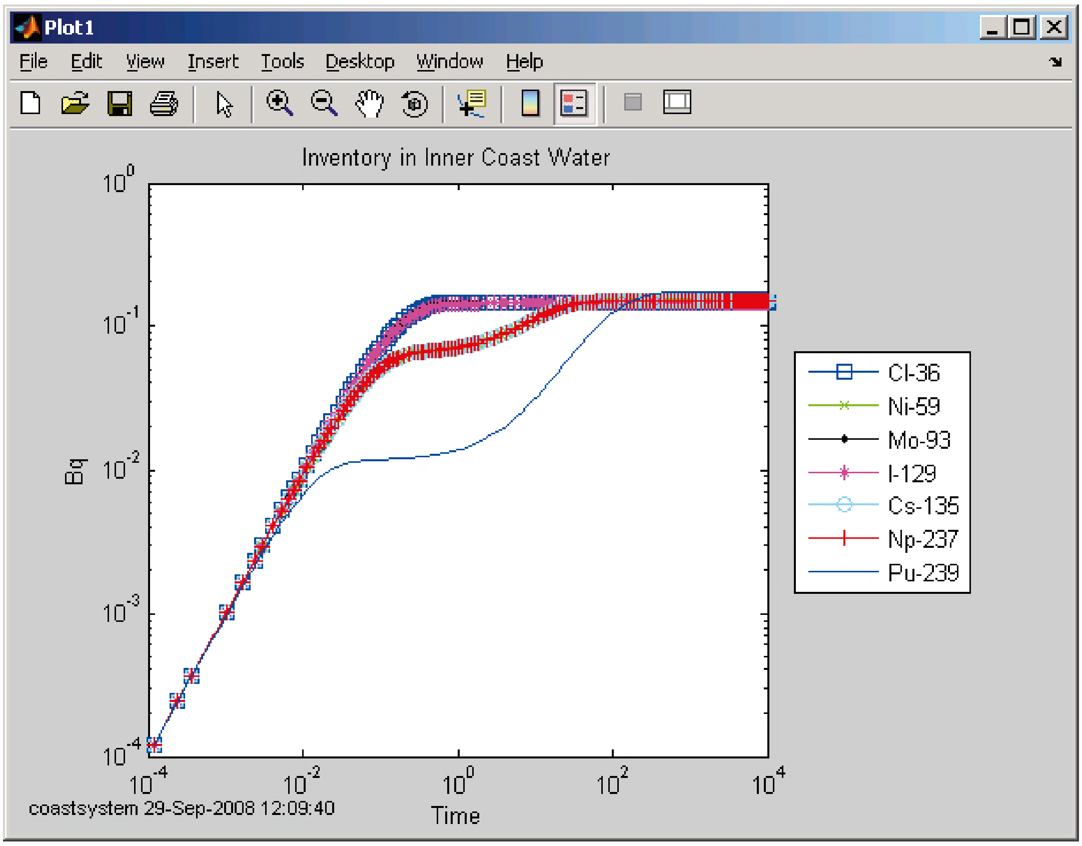


Figure 6-21. *Inventory in Water compartment in Inner Coast sub-system.*

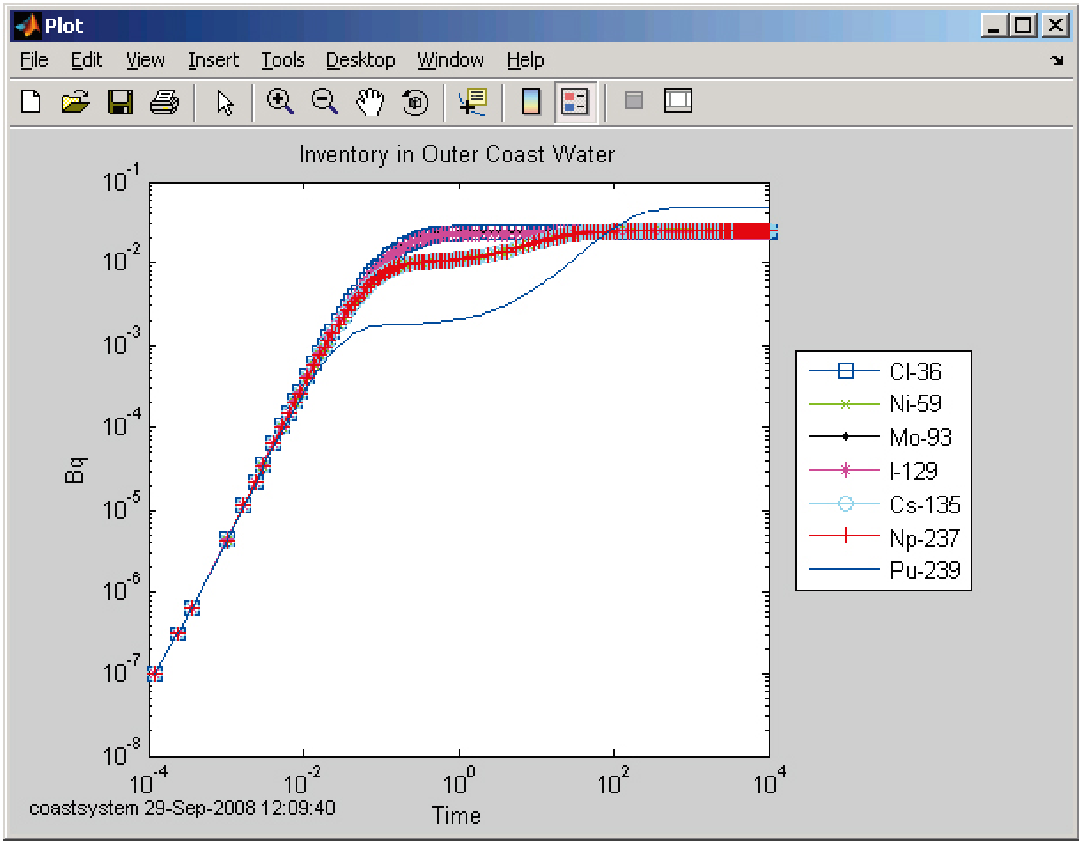


Figure 6-22. *Inventory in Water compartment in Outer Coast sub-system.*

7 Tests and illustrations

Pandora as well as its predecessor Tensit /Jones et al. 2004/ has been benchmarked, tested, and compared with other similar tools. Ecolego /Avila et al. 2005/, a commercial tool based on the MATLAB/Simulink software and utilising similar model approach as Pandora, has undergone several comparisons with other tools /Maul et al. 2003/. We believe that the successful results from these comparisons are implying also confidence in the numerical solution methods in Pandora.

Tensit was compared with several analytical results as well as numerical results from other simulation tools /Robinson et al. 2003/; the comparison is described in /Jones et al. 2004/. Since the basic concepts underlying Pandora have not changed since Tensit the verifications here are only to ensure that quality is maintained after all changes.

As a verification that Pandora still works satisfactory, the test cases used in Chapter 8 in /Jones et al. 2004/ has been carried out again with the updated tool. In each of the comparisons we illustrate how well the Pandora implementation complies with the conceptual model.

All Pandora implementations used in this comparison can be found in the documentation directory of the Pandora software.

7.1 Test of the fundamental Reservoir block

Three test cases were setup and tested in the AMBER 4.5 Verification summary by /Robinson et al. 2003/. These test cases were used as reference for the comparisons in this subsection. The introduced definitions of each respective case are repeated in this text.

These tests are intended to indicate the precision in results when using Pandora's Reservoir block with a given solver and error tolerance level.

7.1.1 SN 2-reference model

This model consists of three compartments (**A**, **B** and **C**) with transport of radionuclides according to Table 7-1.The calculation takes into account the decay of two nuclides: Nuclide 1 decays at a rate of 1e^{-4} y^{-1} and Nuclide 2 at a rate of 1e^{-2} y^{-1}. Input from the source to **A** is for Nuclide 1 only and equals zero except for two time intervals: from 0 to 10 years (1 moles/year) and from 30 to 50 years (2 moles/year). The initial transfer rates are given in Table 7-1. After 40 years, all transfer rates are decreased by a factor of 100. The conceptual representation of the SN 2-reference model is illustrated in Figure 7-1, with the corresponding Pandora implementation illustrated in Figure 7-2.

Simulation was carried out using Simulink default settings and the ode-solver ODE45 with a relative tolerance 1E-7.

The results of the Pandora implementation are compared to the AMBER calculations /Robinson et al. 2003/ after simulation times of 10 and 100 years, as listed in Table 7-2. The results comparing with AMBER shows agreement to four significant figures. Inventories over time in the three reservoirs are plotted using the Result Viewer and illustrated in Figure 7-3.

7.1.2 SN 5-reference model

Reference model SN 5 described in /Robinson et al. 2003/ and /Jones et al. 2004/ consists of three compartments without transfers only integrating each other's inventory. The conceptual representation is illustrated in Figure 7-4 and the implementation in Pandora is shown in Figure 7-5. We only have one nuclide in this model and it decays at a rate of 0.01 per year, but this decay is done in reservoir **A** only. The transfers are both non-depleting and have a rate of 1 per year. That is, **B** calculates the integral of **A**, and **C** calculates the integral of **B**. Initially, there is a unit mass in **A**.

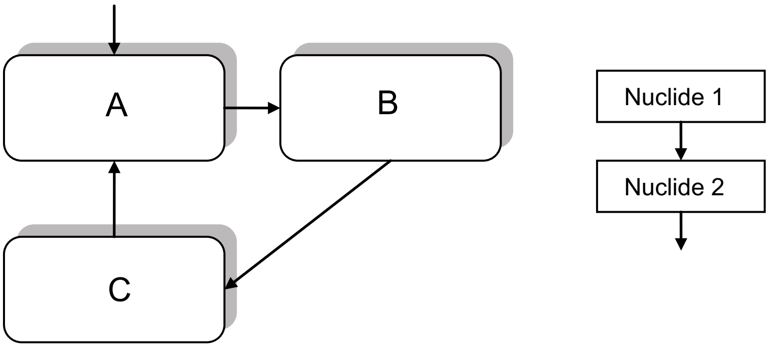


Figure 7-1. *Conceptual representation of the SN 2-reference model.*

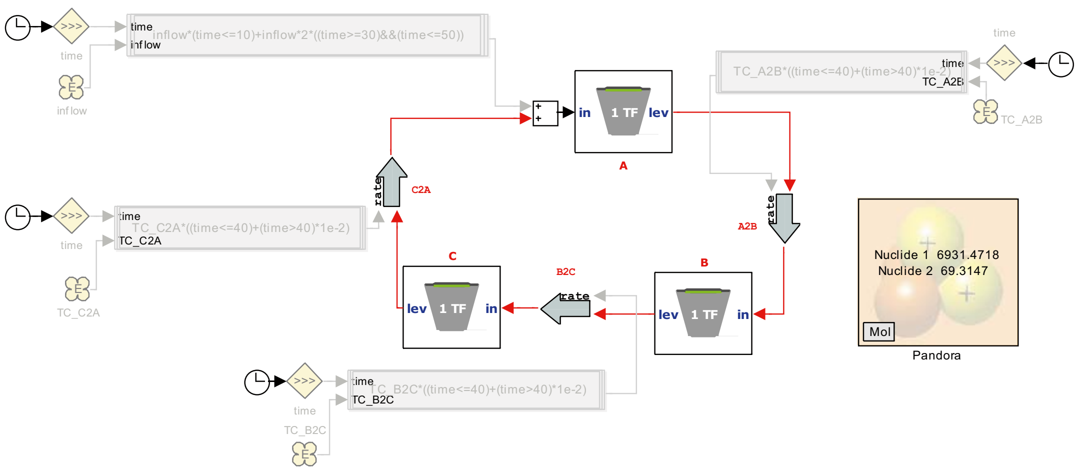

>>>
time
inflow
inflow*(time<=10)+inflow*2*((time>=30)&&(time<=50))
TC_A2B*((time<=40)+(time>40)*1e-2)
TC_A2B
in
1 TF
lev
A
rate
C2A
A2B
TC_C2A*((time<=40)+(time>40)*1e-2)
TC_C2A
C
B2C
B
Nuclide 1 6931.4718
Nuclide 2 69.3147
Mol
Pandora
TC_B2C*((time<=40)+(time>40)*1e-2)
TC_B2C

Figure 7-2. *The SN 2-reference model implemented in Pandora.*

Table 7-1. Initial transfer rates for the SN 2 reference model. After 40 years, all transfer rates decrease by a factor of 100.

From	To	Nuclide 1	Nuclide 2
A	B	0.01	0.001
B	C	0.001	0.1
C	A	0.1	0.1

Table 7-2. Results of the Pandora implementation and run of the SN 2 reference model compared with the results with AMBER, /Robinson et al. 2003/.

Compartment	Nuclide	Time [year]	AMBER [mole]	Pandora [mole]
A	Nuclide 1	10	9.51191	9.511911532
A	Nuclide 2	10	0.00466821	0.004668213
B	Nuclide 1	10	0.481807	0.481807088
B	Nuclide 2	10	0.000137849	0.000137849
C	Nuclide 1	10	0.00128305	0.001283046
C	Nuclide 2	10	2.97307E–5	2.97308E-05
A	Nuclide 1	100	45.5501	45.55009064
A	Nuclide 2	100	0.220230	0.22023007
B	Nuclide 1	100	4.09113	4.091129299
B	Nuclide 2	100	0.0188336	0.018833582
C	Nuclide 1	100	0.0249555	0.024955489
C	Nuclide 2	100	0.00142203	0.001422028

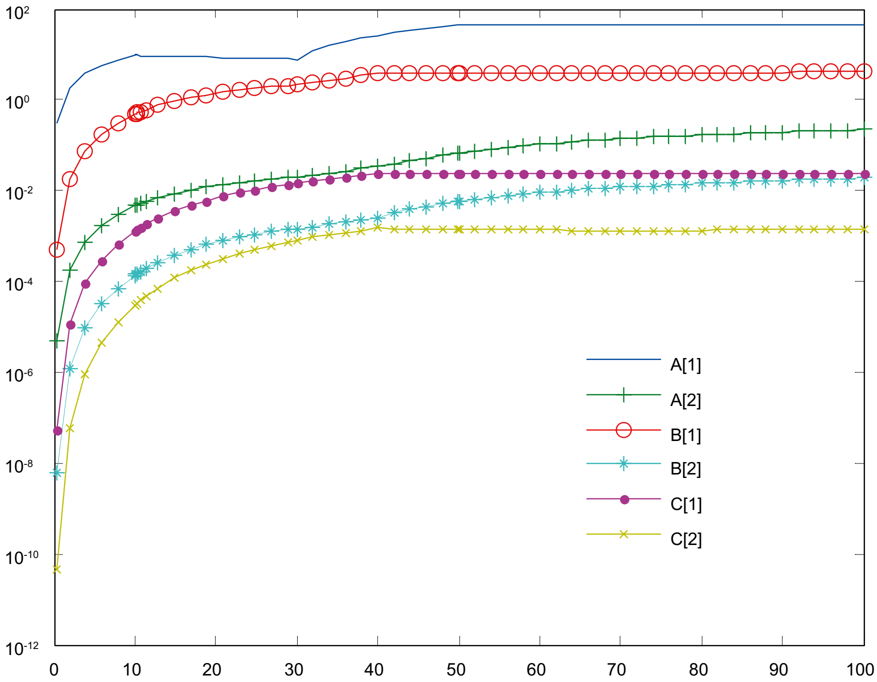


Figure 7-3. *Inventory results from the SN 2-reference model implemented in Pandora.*

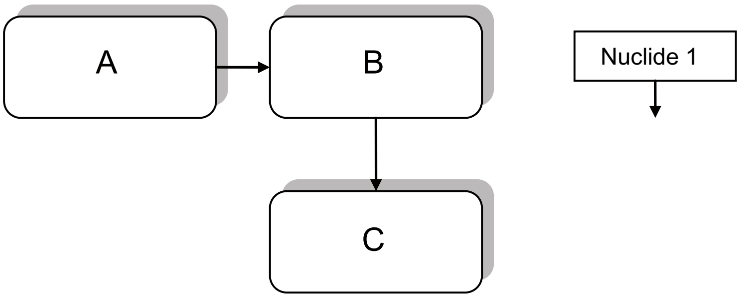


Figure 7-4. *Conceptual representation of the SN 5-reference model.*

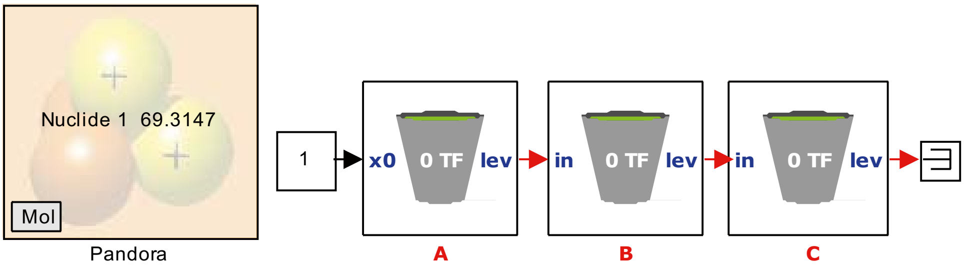


Figure 7-5. *The SN 5-reference model implemented in Pandora.*

The analytical solution to this system is given by

$$A = e^{-\lambda t}$$

$$B = \frac{1-e^{-\lambda t}}{\lambda}$$

$$C = \frac{\mathrm{t}}{\lambda} - \frac{1-e^{-\lambda t}}{\lambda^2}$$

where λ is the decay constant (natural logarithm of 2 divided by the half-life of the radionuclide).

To remove the built in decay computations for reservoir **B** and **C** in the Pandora implementation one has to manually change the decay matrix inside the masked system to be the constant value 0. Results from Pandora implementation simulated using default settings and solver ODE45 are compared with the analytical solution in Table 7-3. As for the SN 2-reference model comparison, the results comparing with analytical solution shows exact agreement to six significant figures with the given solver settings.

Table 7-3. Results of the Pandora implementation and run of the SN 5-reference model compared with analytical solution.

Reservoir	Time [year]	Analytical solution [mole]	Solution obtained with Pandora [mole]
A	0.01	0.999900	0.999900
B	0.01	0.0099995	0.0099995
C	0.01	4.99983E–5	4.99983E-5
A	10	0.904837	0.904837
B	10	9.51626	9.51626
C	10	48.3742	48.3742
A	100	0.367879	0.367879
B	100	63.2121	63.2121
C	100	3,678.79	3,678.79
A	1,000	4.53999E–5	4.540000E-5
B	1,000	99.9955	99.9955
C	1,000	90,000.5	90,000.5

7.1.3 SN 7-reference model

Model reference model SN 7 consists of 12 interconnected compartments in a circular way as seen in the conceptual representation in Figure 7-7.

All interconnected transfer rates are 0.1 per year and the initial amounts of material are one mole in compartment A1, A2, A4, A6, A7 and A10 and zero moles elsewhere. Pandora implementation of SN 7 is illustrated in Figure 7-8. The modelled material (named Nuclide 1) is stable in this test case and therefore we use infinity (Inf) as half-life in the Pandora block.

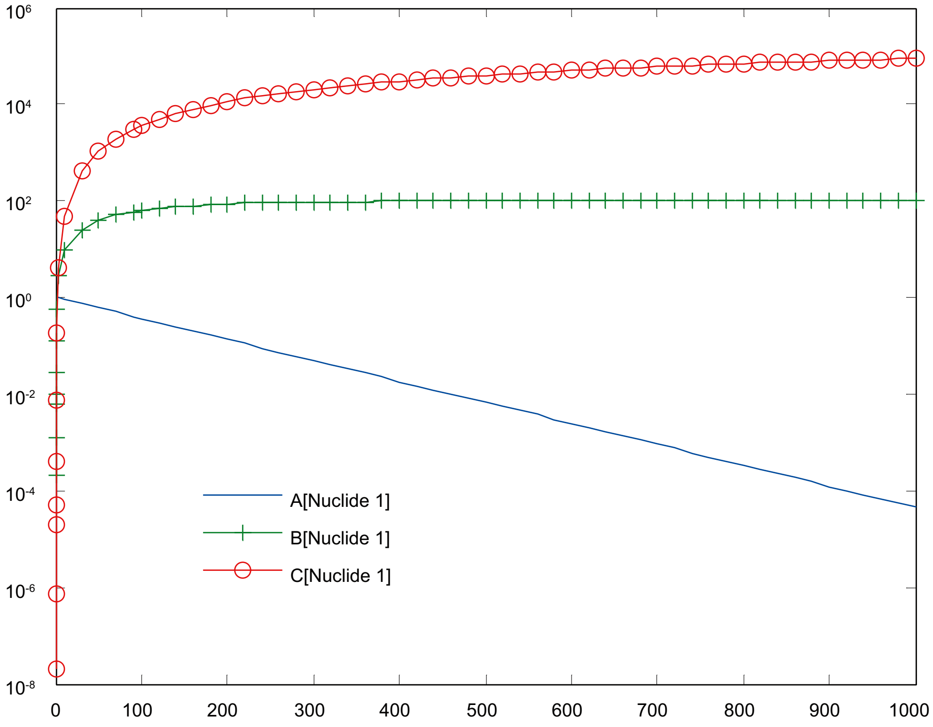


Figure 7-6. Inventory results from the SN 5-reference model implemented in Pandora.

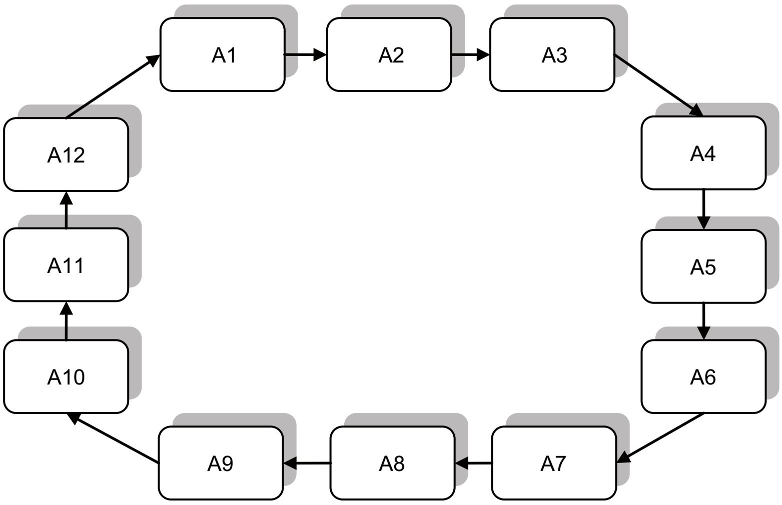


Figure 7-7. *Conceptual representation of the SN 7-reference model.*

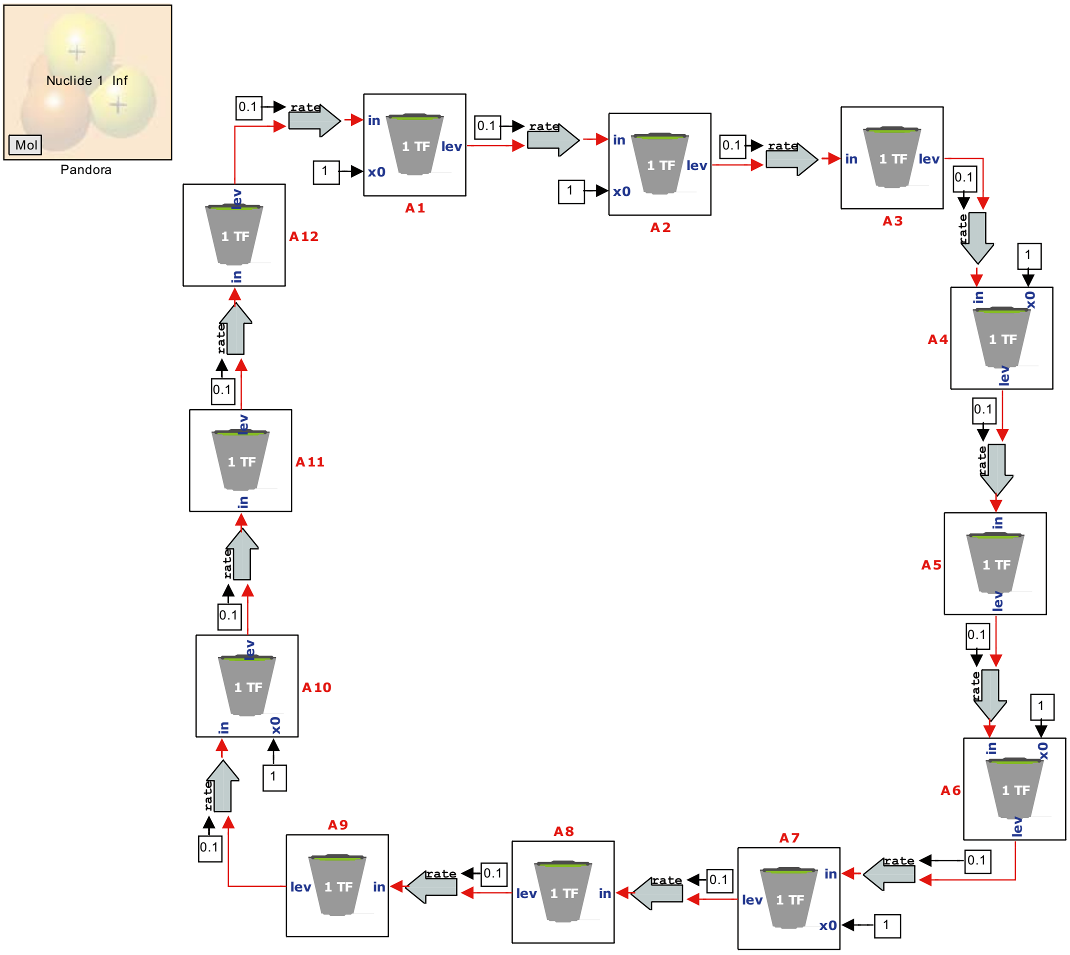


Figure 7-8. *The SN 7-reference model implemented in Pandora.*

Table 7-4 presents analytical results of the SN 7 reference model presented in /Jones et al. 2004/. In Table 7-5 we present the results running the model implemented in Pandora. To achieve the same results we used the solver ODE113 that is designed for problems with stringent error tolerances. In addition, an absolute error tolerance of $1.0{\cdot}10^{-15}$ and a relative tolerance of $1.0{\cdot}10^{-10}$ were used.

As in /Jones et al. 2004/ results from model simulation match perfectly up to six significant digits reported in the analytical values table. The Pandora implementation of the SN 7-reference model can be found in the documentation folder of Pandora Tools.

Table 7-4. Analytical solution presented in /Jones et al. 2004/.

Reservoir	Amount	[moles]			
	t=20	t=40	t=60	t=80	t=100
A1	0.331448	0.393231	0.493590	0.545518	0.541057
A2	0.500565	0.382391	0.450256	0.521544	0.542698
A3	0.578488	0.421262	0.422288	0.489768	0.532723
A4	0.598713	0.483640	0.421521	0.460279	0.513205
A5	0.544824	0.530970	0.445906	0.443184	0.489534
A6	0.533186	0.548898	0.481307	0.443560	0.468849
A7	0.634764	0.561520	0.514184	0.459365	0.457322
A8	0.647070	0.584441	0.540759	0.484075	0.457839
A9	0.491510	0.589493	0.561193	0.510966	0.469434
A10	0.419087	0.556907	0.570820	0.534847	0.488415
A11	0.400650	0.503189	0.562832	0.551199	0.509930
A12	0.319694	0.444059	0.535343	0.555695	0.528994

Table 7-5. Solution obtained using implementation in Pandora. Results match the analytical solution in Table 7-4.

Reservoir	Amount	[moles]			
	t=20	t=40	t=60	t=80	t=100
A1	0.331448271	0.393230971	0.49359003	0.545518434	0.541056641
A2	0.500565256	0.38239074	0.450255835	0.521544153	0.542697563
A3	0.57848791	0.421261708	0.422288337	0.489768055	0.53272315
A4	0.598713022	0.483640014	0.421520663	0.460278929	0.513205495
A5	0.544823535	0.53096979	0.445906495	0.443183894	0.489534434
A6	0.533186179	0.548898297	0.481307364	0.443559684	0.468849108
A7	0.634764394	0.56152031	0.514184209	0.459365047	0.457322185
A8	0.647069938	0.584440738	0.540759266	0.484074707	0.457838626
A9	0.49151035	0.589492829	0.561193004	0.510965723	0.469433622
A10	0.419087035	0.556906794	0.570819897	0.53484741	0.488415237
A11	0.400649902	0.503188929	0.56283178	0.551198735	0.509929965
A12	0.319694209	0.444058879	0.53534312	0.55569523	0.528993974

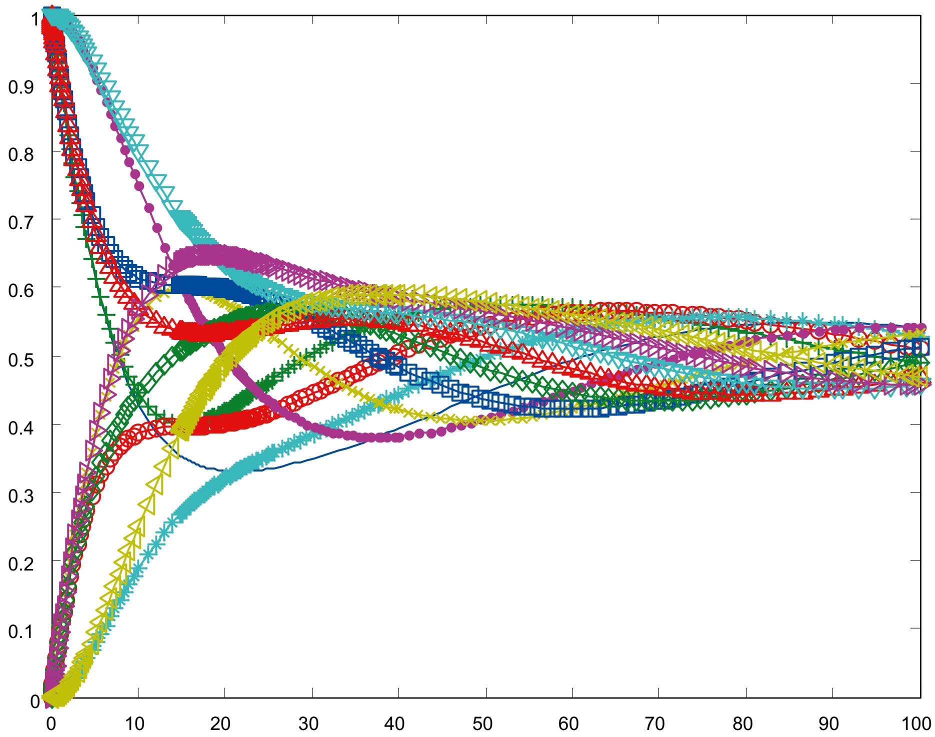


Figure 7-9. *Inventory results from the SN 7-reference model implemented in Pandora (legend was omitted due to the large amount of variables).*

7.2 PSACOIN Level 1B

In the previous section we have compared some deterministic reference models with results from AMBER and analytical solutions with successful results. A big difference between Tensit and the current version of Pandora is the addition of the assessment tool Pandas and the possibility to incorporate probabilistic simulations within an assessment. Therefore we also chose to implement and simulate the PSACOIN Level 1B exercise, /Klos et al. 1993/, as in /Jones et al. 2004/. PSACOIN Level 1B is an international test case from 1993 where eight participating groups compared their results after having simulated a given dose model, both deterministically and probabilistically. In this comparison, we are only interested in the probabilistic result comparison.

As in the previous Tensit comparison, we compare our results with the whole range of results presented by the participating groups. We also present the results from the Tensit comparison separately. The number of realizations used by most participating groups is 1,000. In the Tensit comparison, a significantly higher number of realizations were used to achieve convergence of the mean with its random sampling approach. By contrast, Pandas has the capability to sample input data using Latin Hypercube sampling, which requires fewer realizations to produce stable estimates of the output distribution. The Pandas results using only 1,000 realizations with Latin Hypercube Sampling produced stable estimates of the output, and this is considered sufficient for the current comparison.

A conceptual representation of the three sub-systems of the PSACOIN Level 1B model is illustrated in Figure 7-10. Figure 7-11 illustrates its Pandora implementation counterpart. By comparing the figures, it is clear that the Pandora implementation provides good visual resemblance to the conceptual illustration.

The Pandora model file with the parameter Excel sheet and Pandas assessment file used for this comparison is available in the documentation folder in Pandora Tools.

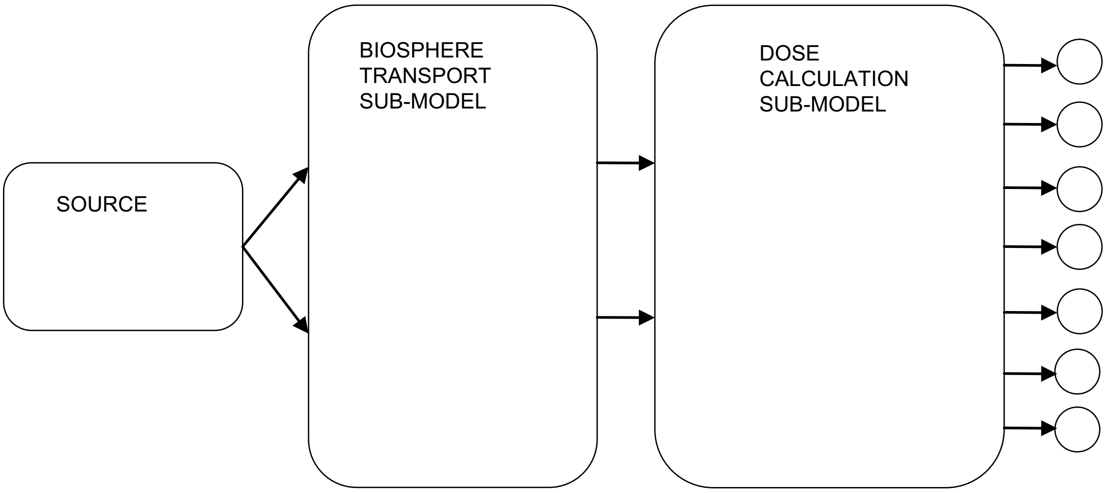


Figure 7-10. *Conceptual representation of the PSACOIN Level 1B.*

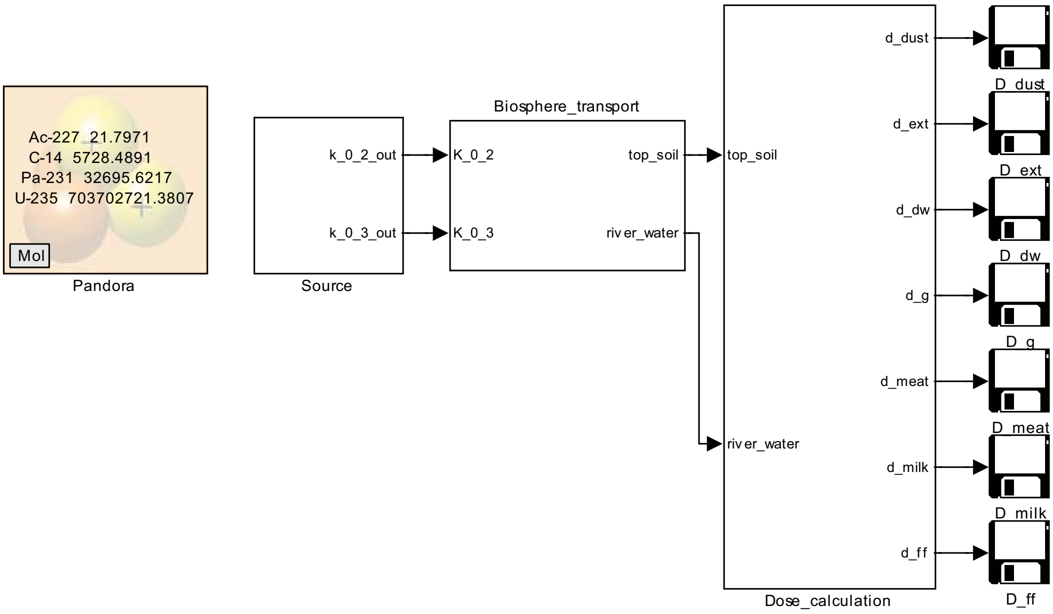


Figure 7-11. *Top level in PSACOIN Level 1B Pandora implementation.*

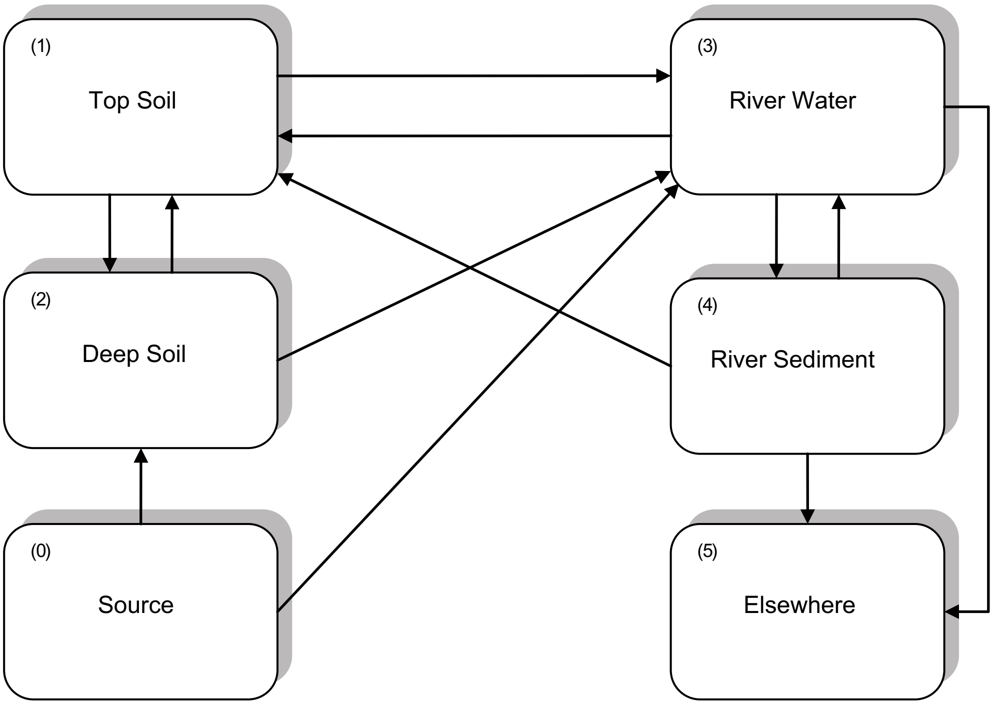


***Figure 7-12.** Conceptual model of the Source and Biosphere Transport Sub-model(s).*

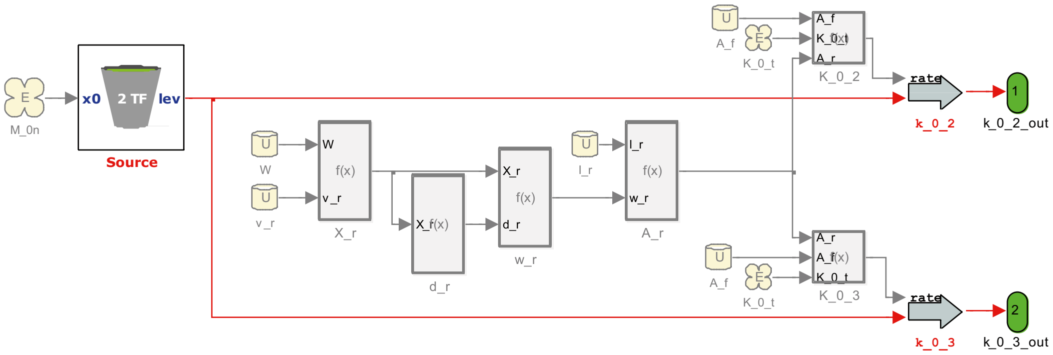


***Figure 7-13.** Source sub-system implemented in Pandora.*

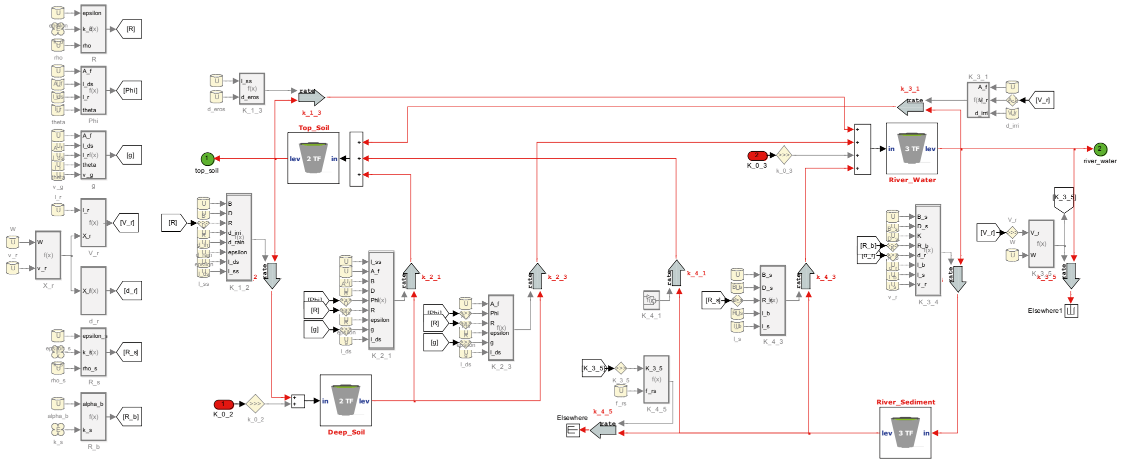


***Figure 7-14.** Biosphere transport sub-system implemented in Pandora.*

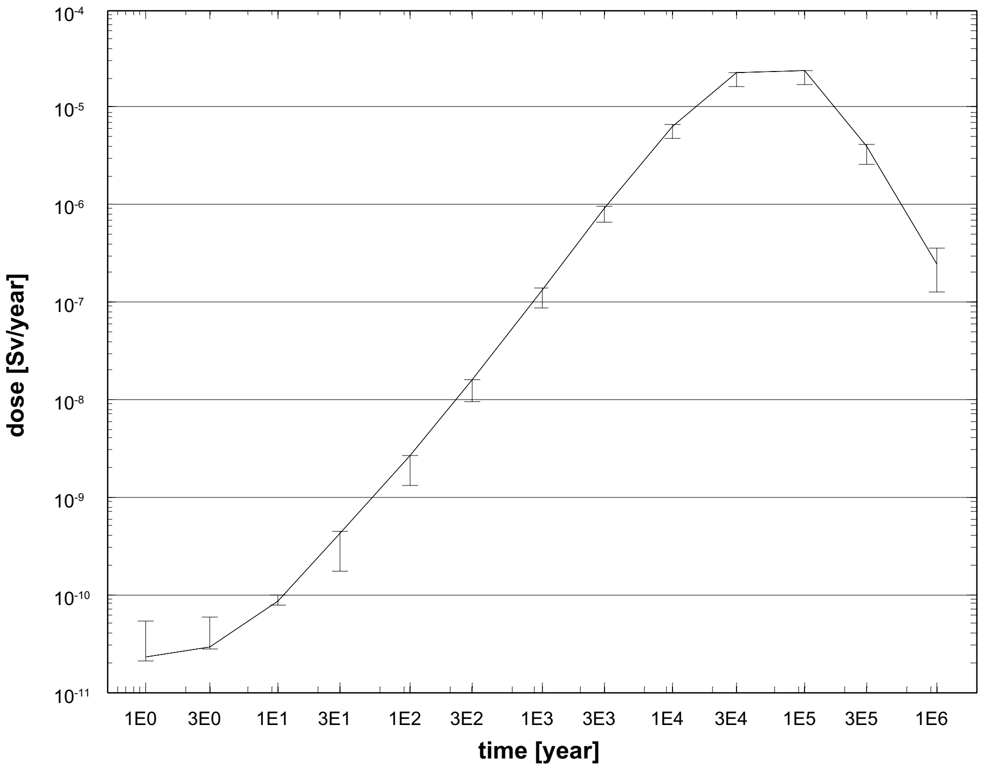


Figure 7-15. Mean total dose from the ^{235}U chain. An interpolated line between data has been added. The staples show the span of results reported from the eight participating groups.

Table 7-6. Mean total dose from ^{235}U chain computed with Pandora compared with Tensit and range of values reported by eight participating test groups. Both Pandora and Tensit results are within the range.

Time [year]	Pandora [Sv/year]	Tensit [Sv/year]	min [Sv/year]	max [Sv/year]
1	$2.3 \cdot 10^{-11}$	$2.1 \cdot 10^{-11}$	$2.10 \cdot 10^{-11}$	$5.40 \cdot 10^{-11}$
3	$2.9 \cdot 10^{-11}$	$2.9 \cdot 10^{-11}$	$2.80 \cdot 10^{-11}$	$5.90 \cdot 10^{-11}$
10	$8.7 \cdot 10^{-11}$	$8.8 \cdot 10^{-11}$	$7.90 \cdot 10^{-11}$	$1.00 \cdot 10^{-10}$
30	$4.3 \cdot 10^{-10}$	$4.3 \cdot 10^{-10}$	$1.70 \cdot 10^{-10}$	$4.40 \cdot 10^{-10}$
100	$2.7 \cdot 10^{-09}$	$2.7 \cdot 10^{-09}$	$1.30 \cdot 10^{-09}$	$2.70 \cdot 10^{-09}$
300	$1.6 \cdot 10^{-08}$	$1.6 \cdot 10^{-08}$	$9.40 \cdot 10^{-09}$	$1.60 \cdot 10^{-08}$
1,000	$1.3 \cdot 10^{-07}$	$1.3 \cdot 10^{-07}$	$8.90 \cdot 10^{-08}$	$1.40 \cdot 10^{-07}$
3,000	$9.2 \cdot 10^{-07}$	$9.3 \cdot 10^{-07}$	$6.60 \cdot 10^{-07}$	$9.50 \cdot 10^{-07}$
10,000	$6.3 \cdot 10^{-06}$	$6.4 \cdot 10^{-06}$	$4.70 \cdot 10^{-06}$	$6.50 \cdot 10^{-06}$
30,000	$2.2 \cdot 10^{-05}$	$2.2 \cdot 10^{-05}$	$1.60 \cdot 10^{-05}$	$2.30 \cdot 10^{-05}$
100,000	$2.3 \cdot 10^{-05}$	$2.2 \cdot 10^{-05}$	$1.70 \cdot 10^{-05}$	$2.40 \cdot 10^{-05}$
300,000	$4.0 \cdot 10^{-06}$	$3.5 \cdot 10^{-06}$	$2.60 \cdot 10^{-06}$	$4.20 \cdot 10^{-06}$
1,000,000	$2.5 \cdot 10^{-07}$	$2.2 \cdot 10^{-07}$	$1.30 \cdot 10^{-07}$	$3.50 \cdot 10^{-07}$

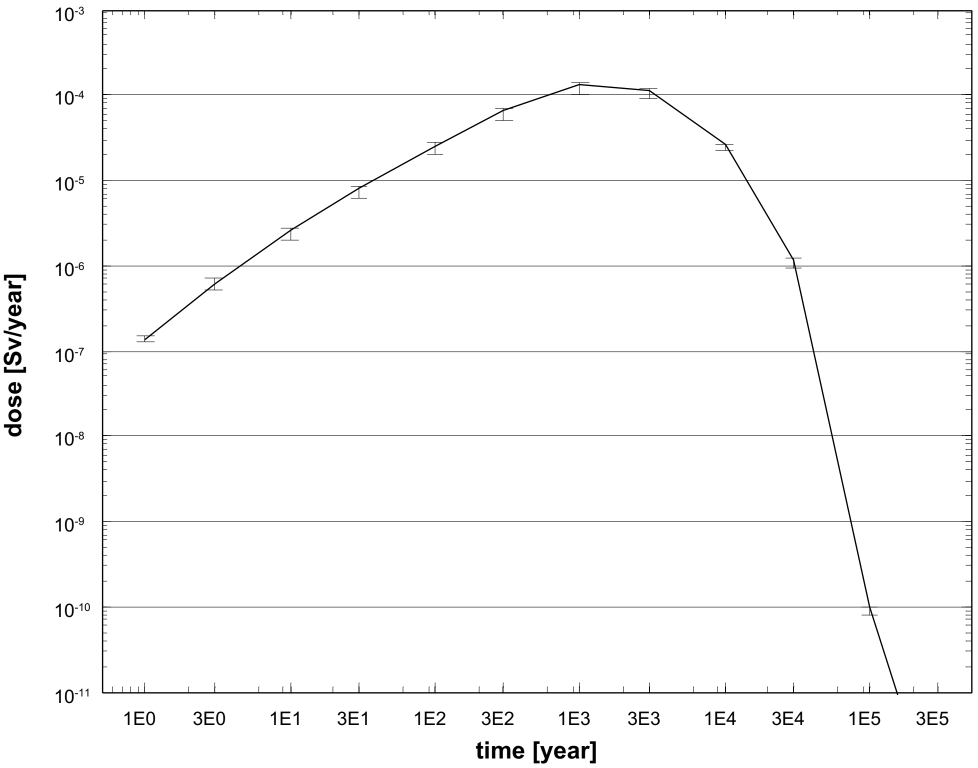


Figure 7-16. *Mean total dose from ^{14}C. An interpolated line between data has been added. The staples show the span of results reported from the eight participating groups.*

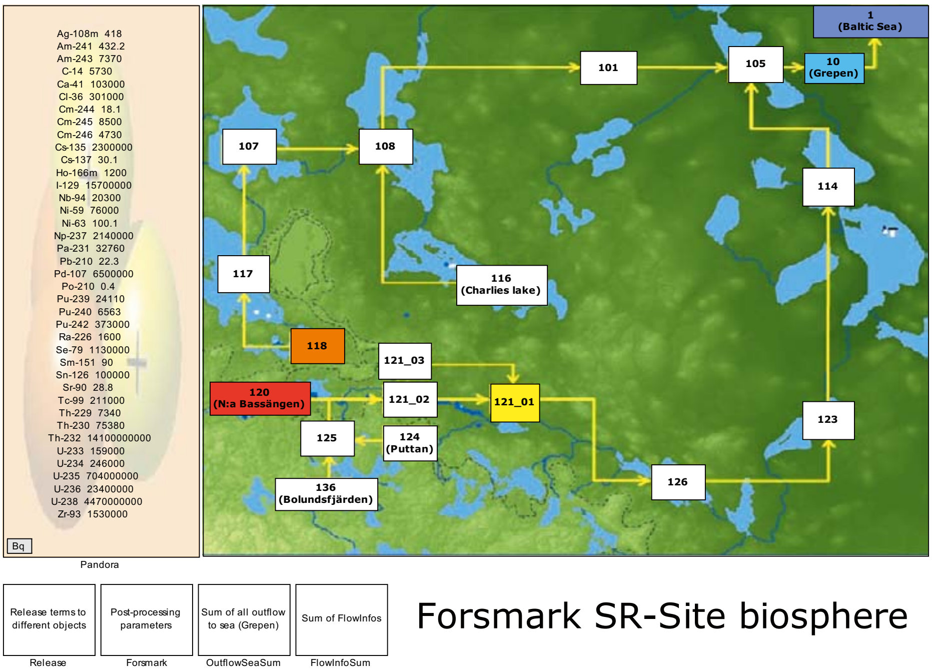
Table 7-7. Mean total dose from ^{14}C computed with Pandora compared with Tensit and range of values reported by eight participating test groups. Pandora results are within the range.

Time [year]	Pandora [Sv/year]	Tensit [Sv/year]	min [Sv/year]	max [Sv/year]
1	$1.3 \cdot 10^{-07}$	$1.5 \cdot 10^{-07}$	$1.30 \cdot 10^{-07}$	$1.50 \cdot 10^{-07}$
3	$6.2 \cdot 10^{-07}$	$6.8 \cdot 10^{-07}$	$5.30 \cdot 10^{-07}$	$7.00 \cdot 10^{-07}$
10	$2.5 \cdot 10^{-06}$	$2.7 \cdot 10^{-06}$	$2.00 \cdot 10^{-06}$	$2.80 \cdot 10^{-06}$
30	$8.0 \cdot 10^{-06}$	$8.5 \cdot 10^{-05}$	$6.20 \cdot 10^{-06}$	$8.70 \cdot 10^{-06}$
100	$2.6 \cdot 10^{-05}$	$2.7 \cdot 10^{-05}$	$2.00 \cdot 10^{-05}$	$2.80 \cdot 10^{-05}$
300	$6.6 \cdot 10^{-05}$	$7.1 \cdot 10^{-05}$	$5.10 \cdot 10^{-05}$	$7.10 \cdot 10^{-05}$
1,000	$1.3 \cdot 10^{-04}$	$1.4 \cdot 10^{-04}$	$1.00 \cdot 10^{-04}$	$1.40 \cdot 10^{-04}$
3,000	$1.1 \cdot 10^{-04}$	$1.2 \cdot 10^{-04}$	$9.30 \cdot 10^{-05}$	$1.10 \cdot 10^{-04}$
10,000	$2.6 \cdot 10^{-05}$	$2.7 \cdot 10^{-05}$	$2.20 \cdot 10^{-05}$	$2.70 \cdot 10^{-05}$
30,000	$1.1 \cdot 10^{-06}$	$1.2 \cdot 10^{-06}$	$9.60 \cdot 10^{-07}$	$1.20 \cdot 10^{-06}$
100,000	$9.7 \cdot 10^{-11}$	$9.8 \cdot 10^{-11}$	$7.80 \cdot 10^{-11}$	$1.00 \cdot 10^{-10}$
300,000	$3.0 \cdot 10^{-13}$	$-1.3 \cdot 10^{-15}$	$7.30 \cdot 10^{-22}$	$5.10 \cdot 10^{-13}$
1,000,000	$3.0 \cdot 10^{-15}$	$1.1 \cdot 10^{-16}$	*	$6.90 \cdot 10^{-15}$

7.3 Forsmark Landscape model

As an illustration of a more complex analysis we present a so called landscape model developed within the SR-Site Biosphere project /Avila et al. 2010/. The model is part of the work performed for assessing the long-term safety for the two candidate sites for a final repository for spent nuclear fuel in Forsmark and in Laxemar. The Forsmark site is today close to the sea but in the future, the shoreline is expected move away from the repository. The landscape model takes this change of ecosystem into account. The landscape is discretised into sub-sections called landscape objects with transport of radionuclides between them; these systems can be seen in the Pandora implementation in Figure 7-17.

Each of the boxes on the map corresponds to a landscape object which itself is a dynamic ecosystem model parameterised to change from being a sea, transferring into a terrestrial object with a lake, which with time will become a mire. A conceptual representation of a landscape object can be seen in Figure 7-18.



Ag-108m	418
Am-241	432.2
Am-243	7370
C-14	5730
Ca-41	103000
Cl-36	301000
Cm-244	18.1
Cm-245	8500
Cm-246	4730
Cs-135	2300000
Cs-137	30.1
Ho-166m	1200
I-129	15700000
Nb-94	20300
Ni-59	76000
Ni-63	100.1
Np-237	2140000
Pa-231	32760
Pb-210	22.3
Pd-107	6500000
Po-210	0.4
Pu-239	24110
Pu-240	6563
Pu-242	373000
Ra-226	1600
Se-79	1130000
Sm-151	90
Sn-126	100000
Sr-90	28.8
Tc-99	211000
Th-229	7340
Th-230	75380
Th-232	14100000000
U-233	159000
U-234	246000
U-235	704000000
U-236	23400000
U-238	4470000000
Zr-93	1530000

Figure 7-17. Top level system of the Forsmark landscape model in SR-Site bio.

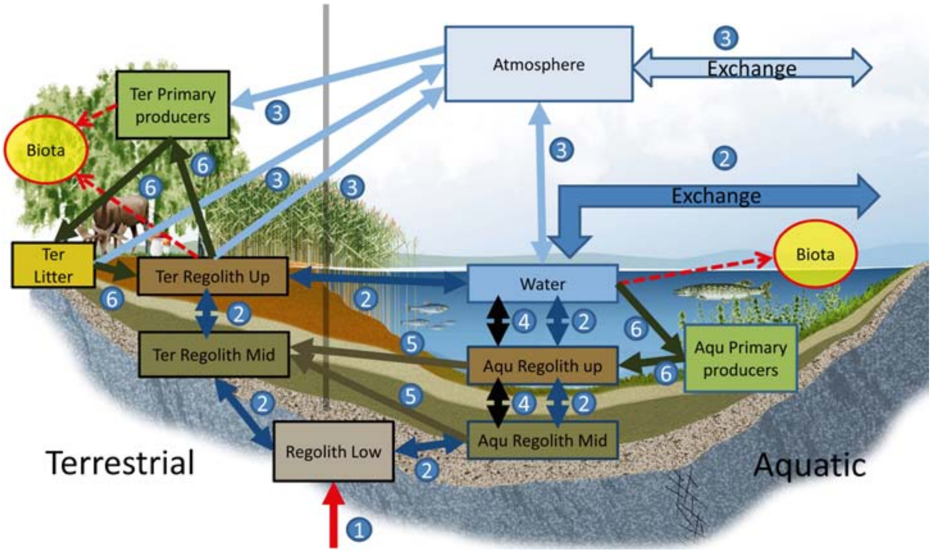


Figure 7-18. Conceptual representation of a landscape object in SR-Site Biosphere.

8 References

SKB's (Svensk Kärnbränslehantering AB) publications can be found at www.skb.se/publications.

Avila R, 2006. The ecosystem models used for dose assessments in SR-Can. SKB R-06-81, Svensk Kärnbränslehantering AB.

Avila R, Bergström U, 2006. Methodology for calculation of doses to man and implementation in Pandora. SKB R-06-68, Svensk Kärnbränslehantering AB.

Avila R, Bergström U, 2007. Methodology for calculation of doses to man and implementation in Pandora. Posiva Working Report 2006-56, Posiva Oy, Finland.

Avila R, Pröhl G, 2008. Models used in the SFR 1 SAR-08 and KBS-3H safety assessments for calculation of ^{14}C doses. SKB R-08-16, Svensk Kärnbränslehantering AB.

Avila R, Broed R, Pereira A, 2003. Ecolego – a toolbox for radioecological risk assessments. In: Proceedings of the International Conference on the Protection of the Environment from the Effects of Ionising Radiation, Stockholm, Sweden, 6–10 October 2003. Vienna: International Atomic Energy Agency.

Avila R, Ekström P-A, Kautsky U, 2006. Development of Landscape Dose Factors for dose assessments in SR-Can. SKB TR-06-15, Svensk Kärnbränslehantering AB.

Avila R, Ekström P-A, Åstrand P-G, 2010. Landscape dose conversion factors used in the safety assessment SR-Site. SKB TR-10-06, Svensk Kärnbränslehantering AB.

Bergström U, Avila R, Ekström P-A, de la Cruz I, 2008. Dose assessments for SFR 1. SKB R-08-15, Svensk Kärnbränslehantering AB.

Bergström U, Nordlinder S, Aggeryd I, 1999. Models for dose assessments. Modules for various biosphere types. SKB TR-99-14, Svensk Kärnbränslehantering AB.

Broed R, 2007. Landscape modelling case studies for Olkiluoto site in 2005–2006. Posiva Working Report 2007-39, Posiva Oy, Finalnd.

Broed R, 2008. Landscape model configuration for biosphere analysis of selected cases in TILA-99 and in KBS-3H safety evaluation, 2007. Posiva Working Report 2007-108, Posiva Oy, Finalnd.

Broed R, Avila R, Bergström U, Hjerpe T, Ikonen A T K, 2009. Biosphere analysis for selected cases in TILA-99 and in KBS-3H safety evaluation, 2007. Posiva Working Report 2007-109, Posiva Oy, Finalnd.

Ekström P-A, Broed R, 2006. Sensitivity analysis methods and a biosphere test case implemented in EIKOS. Posiva Working Report 2006-31, Posiva Oy, Finland.

Jones J, Vahlund F, Kautsky U, 2004. Tensit – a novel probabilistic simulation tool for safety assessments. Tests and verifications using biosphere models. SKB TR-04-07, Svensk Kärnbränslehantering AB.

Karlsson S, Bergström U, Rosén K, 2000. Sensitivity analysis for modules for various biosphere types. SKB R-00-39, Svensk Kärnbränslehantering AB.

Klos R, Sinclair J E, Torres C, Bergström U, Galson D A, 1993. PSACOIN level 1B intercomparison. An international code intercomparison exercise on a hypothetical safety assessment case study for radioactive waste disposal systems, Probabilistic System Assessment Group, Nuclear Energy Agency.

Lindborg T (ed), 2010. Landscape Forsmark – data, methodology and results for SR-Site. SKB TR-10-05, Svensk Kärnbränslehantering AB.

Maul P, Robinson P, Avila R, Broed R, Pereira A, Xu S, 2003. AMBER and Ecolego intercomparisons using calculations from SR97. SKI Report 2003:28, Statens kärnkraftinspektion (Swedish Nuclear Power Inspectorate), SSI Report 2003:11, Statens strålskyddsinstitut (Swedish Radiation Protection Authority).

Robinson P C, Penfold J S S, Little R H, Walke R C, 2003. AMBER 4.5 verification: Summary. Document Reference Number: QRS-1059B-1. [Online]. Available at: http://www.enviros.com/PDF/Services/Users&References_v102e .pdf. [18 August 2003].

SKB, 2008. Site description of Forsmark at completion of the site investigation phase. SDM-Site Forsmark. SKB TR-08-05, Svensk Kärnbränslehantering AB.

SKB, 2009. Site description of Laxemar at completion of the site investigation phase. SDM-Site Laxemar. SKB TR-09-01, Svensk Kärnbränslehantering AB.

SKB, 2011. Long-term safety for the final repository for spent nuclear fuel at Forsmark. Main report of the SR-Site project. SKB TR-11-01, Svensk Kärnbränslehantering AB.

Appendix A

Software requirements

Pandora was developed using MATLAB 7.6.0.324 (R2008a) ©, Simulink 7.1 (R2008a) ©, Windows NT/XP© and Microsoft Excel 2007©.

For the correct operation of Pandora, it is advisable to use the same software releases.

Appendix B

Installing Pandora

To install Pandora one needs to obtain the Pandora Tool folder. The most reliable way is to check out the folder from SKB subversion at: svn://svn.skb.se/projekt/SrSite-Bio/Generic/Simulation/Pandora Tool

After obtaining a copy of Pandora Tool, one needs to follow the steps described in Appendix D to set up the MATLAB path correctly.

Appendix C

Installing Pandas

To install Pandas one needs to obtain the Pandas executable. The most reliable way is to check out the file from SKB subversion at: svn://svn.skb.se/projekt/SrSite-Bio/Generic/Simulation/Pandas/pandas_1.0.32.exe

After obtaining a copy of the Pandas executable, double-click the file and follow the installation steps.

Note; if an earlier version of Pandas is already installed, this version needs to be uninstalled and also a specific folder needs to be removed from the computer to be able to install the later version.

C:\Documents and Settings\<Login Person>\Application Data\Pandas

The folder "Application Data" is hidden so either turn on the "show hidden files" or just write the folder name in the address bar.

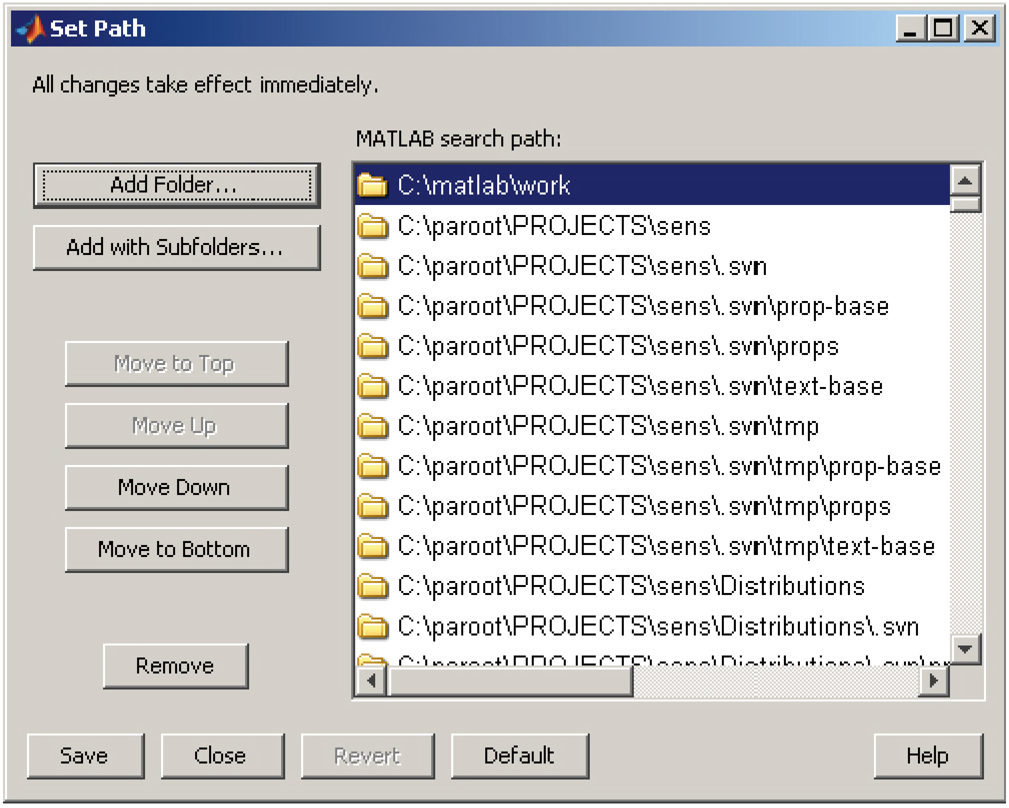
Appendix D

Set the MATLAB path

For MATLAB and Simulink to be able to use the Pandora block library and other features the MATLAB path needs to be set up correctly. The MATLAB path needs to point at the Pandora Tool folder as well as its underlying folders.

To do this one need to choose 'Set Path…' in File menu in MATLAB.

In the Set Path dialog that appears one needs to click the button 'Add with Subfolders…' and choose the folder Pandora Tool. Click 'Save' and then 'Close' and your MATLAB Path has been set.



All changes take effect immediately.

MATLAB search path:

```
C:\matlab\work
C:\paroot\PROJECTS\sens
C:\paroot\PROJECTS\sens\.svn
C:\paroot\PROJECTS\sens\.svn\prop-base
C:\paroot\PROJECTS\sens\.svn\props
C:\paroot\PROJECTS\sens\.svn\text-base
C:\paroot\PROJECTS\sens\.svn\tmp
C:\paroot\PROJECTS\sens\.svn\tmp\prop-base
C:\paroot\PROJECTS\sens\.svn\tmp\props
C:\paroot\PROJECTS\sens\.svn\tmp\text-base
C:\paroot\PROJECTS\sens\Distributions
C:\paroot\PROJECTS\sens\Distributions\.svn
```

Figure D-1. Set Path dialog in MATLAB.