CNWRA *A center of excellence in earth sciences and engineering*

A Division of Southwest Research Institute™
6220 Culebra Road • San Antonio, Texas, U.S.A. 78228-5166
(210) 522-5160 • Fax (210) 522-5155

March 5, 2002
Contract No. NRC-02-97-009
Account No. 20.01402.762

U.S. Nuclear Regulatory Commission
ATTN: Mr. James Firth
Office of Nuclear Material Safety and Safeguards
Division of Waste Management
Performance Assessment and High-Level Waste Integration Branch
Mail Stop 7C–18
Washington, DC 20555

Subject: Graphical Post-Processor Version 1.01 for the Total-system Performance Assessment (TPA) Version 4.0 Code

Dear Mr. Firth:

Attached is a CD containing Version 1.01 of the Graphical Post-Processor developed by Bayesian Systems, Inc. Also attached is a copy of the user's manual. The Graphical post-processor supports visualization of specific data generated with the TPA code Version 4.0. By plotting relevant TPA output variables, the graphical post-processor can be used to quickly assess the influence of TPA input parameters on relevant variables, and the interrelation among these variables. The graphical post-processor can be used to visualize the propagation of uncertainty by various system components. A detailed technical description of the application is presented in the user's manual.

Besides containing the graphical post-processor application, the attached CD includes needed additional software for the application to run in the Windows NT 4.0 operating system. It also includes sample data generated with the TPA Version 4.0 Code, and electronic copies of the user's manual in WordPerfect 8.0 and Adobe 4.0 PDF formats.

If you have any questions related to the graphical post-processor application or the user's manual please contact Dr. Osvaldo Pensado at (210) 522-6084.

Sincerely yours,

Gordon Wittmeyer

Gordon Wittmeyer, Ph.D.
Manager, Performance Assessment

GWW/cw
Enclosure

cc: J. Linehan, B Meehan, D DeMarco, E. Whitt, J Greeves, W. Reamer, T. Essig, J. Schlueter, S Wastler, K Stablein, A Campbell, B. Leslie, R Codell, D. Esh, C Grossman, R K. Johnson, T. McCartin, C. McKenney, J Peckenpaugh, M Rahimi, M. Thaggard, J. Andersen, H. Arlt, J. Ciocco, W. Ford, D Galvin, L Hamdan, A. Henry, A -B Ibrahim, B Jagganath, W. Dam, W. Patrick, CNWRA Directors, CNWRA Element Managers, T. Nagy (SwRI Contracts), P. Maldonaldo, O. Pensado

Washington Office • Twinbrook Metro Plaza #210
12300 Twinbrook Parkway • Rockville, Maryland 20852-1606

USER'S GUIDE TO VERSION 1.0 OF THE GRAPHICAL POST-PROCESSOR FOR TOTAL-SYSTEM PERFORMANCE ASSESSMENT (TPA) VERSION 4.0

February 2002

Developed by
Bayesian Systems, Inc.

Center for Nuclear Waste Regulatory Analyses
Southwest Research Institute
San Antonio, Texas

SOFTWARE LICENSE NOTICE

TECHNICAL SUPPORT

For technical support, contact:

Dr. Osvaldo Pensado
CNWRA
Southwest Research Institute
6220 Culebra Road
San Antonio, TX 78238

HARDWARE REQUIREMENTS

Hardware running the Microsoft™ Windows NT operating system

TABLE OF CONTENTS

Section Page

SOFTWARE LICENSE NOTICE ii
TECHNICAL SUPPORT ii
HARDWARE REQUIREMENTS ii
OVERVIEW v

1 INSTALLATION PROCEDURES 1-1
1.1 Installing the Java™ Runtime Environment 1-1
1.1.1 Supported Platforms 1-1
1.1.2 Installing the Java™ Runtime Environment for Nondevelopers 1-1
1.1.2.1 Obtaining the Java™ Runtime Environment 1-1
1.1.2.2 Running the Java™ Runtime Environment Install Program 1-2
1.1.3 Installing the Java™ Development Kit for Developers 1-2
1.1.3.1 Obtaining the Java™ Development Kit 1-2
1.1.3.2 Running the Java™ Development Kit Install Program 1-2
1.2 Installing Libraries Required by Graphical Post-Processor 1-3
1.2.1 OpenGL 1-3
1.2.2 Java3D™ 1-3
1.2.3 VisAD 1-4
1.2.4 Colt 1-6
1.3 Installing the Graphical Post-Processor 1-6
1.3.1 Graphical Post-Processor Distribution Files 1-6

2 RUNNING THE GRAPHICAL POST-PROCESSOR 2-1
2.1 Configuring Properties Files 2-1
2.2 Running the Program 2-2
2.2.1 Selecting TPA Data for Graphic Display 2-3
2.2.2 Plot Control Window 2-4
2.2.2.1 Sample Case of Graphic Display 2-6
2.2.3 View Control Window 2-8
2.2.3.1 Create or Add a New View 2-8
2.2.3.2 Delete a View 2-9
2.2.3.3 Replace a View 2-9
2.2.4 Displaying Several Variables on the Same Plot 2-9
2.2.5 Influence of Parameters on Variables 2-10
2.2.6 Influence of Variables on Variables 2-13
2.2.7 Scatter Plot for the Peak Dose 2-14
2.2.8 Box and Whisker Plots 2-14
2.2.9 Snap as a JPG or CSV File 2-16
2.3 Sample Plots 2-17

TABLE OF CONTENTS (continued)

Section Page

APPENDIX A—OBTAINING THE JAVA™ FROM THE INTERNET
APPENDIX B—SOFTWARE MAINTENANCE
APPENDIX C—STRATEGY FOR THE SELECTION OF COLOR
APPENDIX D—TOTAL-SYSTEM PERFORMANCE ASSESSMENT (TPA) FILES

OVERVIEW

The Graphical Post-Processor supports visualization of data generated with the Total-system Performance Assessment (TPA) Version 4.0 code. By plotting relevant TPA output variables, the graphical post-processor is intended as a tool for the quick analysis of the influence of TPA input parameters on the relevant variables, and the interrelation among these variables. The graphical post-processor is also intended as a tool for the analysis of uncertainty propagation.

This manual includes installation procedures of the graphical post-processor and basic instructions for the graphic display of TPA data, as well as highlights of available features of the graphical post-processor. The appendices contain information useful for software maintenance.

1 INSTALLATION PROCEDURES

Summary

This section describes installation of the graphical post-processor Version 1.0. The graphical post-processor requires Java™ and third-party libraries [OpenGL, Java3D™, Visualization for Alogorithm Development (VisAD), and Colt], which are on the graphical post-processor distribution compact disc (CD) and may be obtained from the internet. It is best if OpenGL, Java3D™, and VisAD are installed in that sequence. Section 1.1 describes the installation of Java™, Section 2.1, the required libraries, and Section 3.1, the graphical post-processor itself. To avoid software conflicts, it is recommended to install only the Java™ Runtime Environment or the Java™ Development Kit, but not both.

1.1 Installing the Java™ Runtime Environment

1.1.1 Supported Platforms

The graphical post-processor is designed to run on any platform supporting Java™ 2 and Java3D™. The current target for graphical post-processor Version 1.0 is the Microsoft™ Windows NT platform, but the graphical post-processor is expected to perform well in other platforms such as Windows 9x, Windows 2000, Solaris, and Linux RedHat 6.1. Support of the Java3D™ library by the operating system platform allows the possibility to extend the graphical post-processor for the interactive analysis of multidimensional data [currently, the graphical post-processor does not display any three-dimensional (3D) functionality].

The Java™ 2 Platform, Standard Edition provides a complete and secure foundation for building and deploying applications for personal computers and workgroup servers. The Java™ 2 Platform, Standard Edition, is available in the Java™ 2 Development Kit and in the Java™ 2 Runtime Environment.

The Java 2™ Platform, Standard Edition, is needed to run the graphical post-processor application. Version 1.3 of Java™ 2 from Sun is available in the installation CD. If already installed, the version of Java™ 2 can be found in the Production Release of the Java™ Runtime Environment. For Java™ developers, it is recommended to install the Java™ Development Kit, available on the graphical post-processor distribution CD and downloadable from the Internet. The Java3D™ package must be installed in the same directory as the Java™ Runtime Environment or the Java™ Development Kit. Solaris users may need to install system patches for Java™ 2 to work correctly. To avoid software conflicts, it is recommended to install only the Java™ Runtime Environment or the Java™ Development Kit, but not both.

1.1.2 Installing the Java™ Runtime Environment for Nondevelopers

1.1.2.1 Obtaining the Java™ Runtime Environment

The Java™ Runtime Environment must be installed on the host running the graphical post-processor. The graphical post-processor distribution CD (henceforth referred to as the CD)

contains a copy of the self-extracting Java™ Runtime Environment installation program for use with Windows 9x/2000/NT4 at

java/jre/download/j2re1_3_0-win.exe

The CD directory java/jre contains useful information, including a description of the Java™ Runtime Environment, a data sheet, feature description, downloading instructions, installation troubleshooting, licensing information, and changes for the current release. For additional information, consult java.sun.com and other sources listed in Appendix A.

Copy the file j2re1_3_0-win.exe to a temporary directory on your host.

1.1.2.2 Running the Java™ Runtime Environment Install Program

Do not install the Java™ Development Kit if the Java™ Runtime Environment has been installed. To install the Java™ Runtime Environment, run the Java™ Runtime Environment installation package by executing the file j2re1_3_0-win.exe directly from Microsoft™ Windows or from an MS-DOS prompt console. The installation is self-explanatory. The default path to install the Java™ Runtime Environment is c:\program files\javasoft\jre\1.3. However, it is recommended to create a directory named c:\jre1.3 and install the Java™ Runtime Environment there. In the following sections, the directory where the Java™ Runtime Environment has been installed is referred to as $JAVA_HOME. After installation, two new directories will be located at $JAVA_HOME named bin and lib.

1.1.3 Installing the Java™ Development Kit for Developers

1.1.3.1 Obtaining the Java™ Development Kit

The Java™ Development Kit includes tools for building Java™ applications and the Java™ Runtime Environment. A copy of the Java™ Development Kit installation program for Windows 9X/2000/NT4 is available on the CD at

java/jdk/download/jdk130.exe

For other environments, see Appendix A. Copy the file to a temporary directory on a volume accessible to the host on which Java™ will be installed.

1.1.3.2 Running the Java™ Development Kit Install Program

Do not install the Java™ Development Kit if the Java™ Runtime Environment has been installed. To install the Java™ Development Kit, run the Java™ Development Kit installation package by executing the jdk130.exe directly from Microsoft™ Windows or from an MS-DOS prompt console. The installation is self-explanatory. It is recommended to create a directory named c:\jdk1.3 and install the Java™ Development Kit there. In the following sections, the directory where the Java™ Development Kit has been installed is referred to as $JAVA_HOME.

1.2 Installing Libraries Required by Graphical Post-Processor

1.2.1 OpenGL

For Windows 9x platforms, there is a copy of OpenGL on the CD at

libraries/opengl95.exe

Also, this file is freely available from

ftp://ftp.microsoft.com/Softlib/MSLFILES/OPENGL95.EXE

OpenGL is already part of the Microsoft™ Windows NT operating system and does not require installation.

OpenGL for Sparc Solaris is freely available at

http://www.sun.com/solaris/opengl/

OpenGL support for other platforms can be found at

http://opengl.org

1.2.2 Java3D™

Java3D™ must be installed in the same directory selected as the $JAVA_HOME directory. Java3D™ is a set of classes for writing 3D graphic applications and 3D applets. The VisAD library in use by graphical post-processor requires Java3D™ Version 1.1.X; thus, it is necessary to install a compatible version. Do not install the newly released Version 1.2 of Java3D™. There are several incompatibilities with this release, and only the 1.1.X Version is supported by graphical post-processor. For Microsoft™ Windows, there is an installation package for Java3D™ on the CD at

libraries/java3d/java3d1_1_2-win32-opengl-sdk.exe

For Microsoft™ Windows and Solaris (SPARC), the 1.1.X Version can be downloaded from Sun MicroSystems. Install Java3D™ in the same directory where Java™ 2 was installed. Java3D™ requires OpenGL. To install Java3D™, run the file java3d1_1_2_win32_opngl_skd.exe, follow the instructions, and select $JAVA_HOME as the host directory for Java3D™. After installation, the directories demo and jre will be added to the $JAVA_HOME directory.

NOTE:

On Solaris system, there is currently an incompatibility between Java3D™ Version 1.1.X and OpenGL 1.2: moving or resizing a window will cause a program to crash. This bug has been reported to Sun. Until the bug is fixed, OpenGL 1.1.2 must be used, rather than OpenGL 1.2.

Solaris users may need to install Java3D™ system patches. See the documentation accompanying the Java3D™ release for a list of required patches. There are extensive documentation and tutorials on Java3D™ at

http://java.sun.com/java-media/3D/

1.2.3 VisAD

The VisAD library is available on the CD at

libraries/visad/download/visad.jar

This file should be copied to the Java™ Extensions directory. If the Java™ Runtime Environment was installed, the Extensions directory is

$JAVA_HOME/lib/ext

If the Java™ Development Kit was installed, the Java™ Extensions directory is

$JAVA_HOME/jre/lib/ext

Do not expand the archive VisAD.jar, simply copy it into the Java™ Extensions directory. Documentation, examples, source, and a tutorial for VisAD are available on the CD at

libraries/visad/download/v-doc.jar

libraries/visad/download/v-eg.jar

libraries/visad/download/v-src.jar

libraries/visad/download/v-tutor.jar

These files are not needed by VisAD but are included on the CD for reference purposes. The VisAD library was created by programmers of the Space Science and Engineering Center Visualization Project at the University of Wisconsin-Madison Space Science and Engineering Center and by programmers at the Unidata Program Center. VisAD is a Java™ class library for interactive and collaborative visualization and analysis of numerical data. References listed below are in pairs; the first is an on-line reference, and the second points to the location of the document on the CD.

VisAD Resources

License

VisAD is licensed as part of the GNU Public License at the Free Software Foundation
http://www.fsf.org/
libraries/visad/top-doc/GNU-Public-License.txt

The home page, and local copy, for VisAD are at
http://www.ssec.wisc.edu/~billh/visad.html
libraries/visad/top-doc/v-home.htm

Frequently asked questions
http://www.ssec.wisc.edu/~curtis/visad-faq.html
local copy VisAD 2.0 FAQ
libraries/visad/top-doc/faq.htm

Downloadable S/W download directory
ftp://www.ssec.wisc.edu/pub/visad-2.0/visad.jar
libraries/visad/download/visad.jar

Installation instructions
http://www.ssec.wisc.edu/~billh/README
libraries/visad/top-doc/readme.txt

Developer's guide
http://www.ssec.wisc.edu/~billh/guide.html
libraries/visad/top-doc/guide.htm

Tutorial, browsable
http://www.ssec.wisc.edu/~billh/tutorial/index.html
libraries/visad/tutorial/index.html

Tutorial, downloadable
ftp://www.ssec.wisc.edu/pub/visad-2.0/vistutor.jar
libraries/visad/download/v-tutor.jar

Application programming interface, html
http://brutus.ssec.wisc.edu/~dglo/visad/
libraries/visad/docs/index.html

Application programming interface, downloadable
ftp://www.ssec.wisc.edu/pub/visad-2.0/visad_doc-2.0.jar
libraries/visad/download/v-doc.jar

Examples, downloadable
ftp://www.ssec.wisc.edu/pub/visad-2.0/visad_examples.jar
libraries/visad/download/v-eg.jar

Mailing list archive
http://ww.unidata.ucar.edu/staff/russ/visad/threads.html
libraries/visad/mail/thread.html

Source code
ftp://www.ssec.wisc.edu/pub/visad-2.0/visad_src-2.0.jar
libraries/visad/download/v-doc.jar

1.2.4 Colt

The Colt library is available on the CD at

libraries/colt/download/colt1.0.1.zip

The file colt1.0.1.zip should be expanded to a temporary directory, and the contained colt.jar file must be copied to the Java™ Extensions directory, as done with VisAD.

The Colt web site is at

http://tilde-hoschek.home.cern.ch/~hoschek/colt/index.htm

A copy of this site can be found on the CD at

libraries/colt/colt-home.htm

1.3 Installing the Graphical Post-Processor

1.3.1 Graphical Post-Processor Distribution Files

The compiled graphical post-processor code is on the CD at

gpp\yy-mm-dd. The directory yy-mm-dd represents the graphical post-processor version by date (e.g., 01-02-01 or 00-10-2). The latest version on CD should be selected.

The graphical post-processor classes must be copied to a local or a network volume. Use of a network volume facilitates distribution of software updates but requires operability of the network for the Java™ programs to run and could slow performance. The graphical post-processor classes are located on the CD at gpp\yy-mm-dd.

Create a directory in the local or network hard-drive for (e.g., c:\gpp\yy-mm-dd). In the following discussion, this directory is referred to as $GPP_HOME. Copy the CD contents of gpp\yy-mm-dd into $GPP_HOME.

To copy the graphical post-processor classes from the CD into $GPP_HOME, at an MS-DOS prompt type

```
xcopy e:\gpp\yy-mm-dd\* c:\gpp\yy-mm-dd\ /D /S
```

In the previous MS-DOS command, yy-mm-dd must be substituted by the most recent version, and e and c by the appropriate letters referring to the CD-ROM and the hard-drive, respectively. Substitute also c:\gpp\yy-mm-dd\ by the path selected as the $GPP_HOME.

Alternatively, the contents of gpp\yy-mm-dd\ could be dragged into $GPP_HOME using the Windows Explorer interface.

Graphical post-processor source code documentation in HTML format is available on CD in the folder api-doc. This documentation does not need to be copied to the hard drive. It is provided only for reference purposes.

The graphical post-processor classes are specified in the documentation contained in the api-doc directory. Consult the document index.html as a starting point. In this release, the software documentation for the graphical post-processor is integrated with the documentation of the parts of the VisAD library it uses. The documentation is not yet integrated with the Colt documentation.

The source code for graphical post-processor is available on the CD, distributed throughout all *.java files located at

> gpp\yy-mm-dd\, where yy-mm-dd represents the most recent version. Check the CD contents for the most recent version.

2 RUNNING THE GRAPHICAL POST-PROCESSOR

2.1 Configuring Properties Files

Property files are used to configure the output of the graphical post-processor application. Property files are located in the $GPP_HOME directory. They can be modified to customize the graphical post-processor output. The contents of these property files are explained next.

gpp.properties

- *which_subarea* controls the subarea to be analyzed. This line is aimed at controlling the display of radionuclide release rates. If this line is missing in the file, the total number of subareas will be considered for the display of release rates. To indicate a particular subarea, enter the number of the subarea after the = sign with no spaces; for example,

  which_subarea=2

  This line can be commented out by typing the pound symbol, #, at the beginning of the line (e.g., # which_subarea=2).

- *num_realizations* controls how many realizations are to be included in the graphic displays. Entering a number greater or equal than the total number of Total-system Performance Assessment (TPA) realizations will force consideration of all of the realizations. It is recommended to select of a small number for faster performance; for example,

  num_realizations=20

- *param_percentiles* controls the default percentiles to be displayed in the highlight of cross-correlations between parameters and variables and among variables; for example,

  param_percentiles=10,50,100

- *nuclide_of_interest* is used to display release rates on a particular nuclide; for example,

  nuclide_of_interest=Cm245

  causes the graphical post-processor to display release rates of ^{245}Cm. Alternatively, typing

  nuclide_of_interest=none

  will cause all of the radionuclides to be considered in the display of release rates. See Section 2.2.8 for the list of 20 radionuclides with release rates tracked by the TPA Code Version 4.0.

variable_abbrevs.properties

This file allows the user to modify the names of variables to appear on picklists and on graphs in the graphical post-processor. Short variable names display more legibly on the Y-axis of the plots. Underscores are required for Java™ but do not show up in most places in the application. Do not use blanks anywhere in the lines. For example, using

Water_Hitting_Waste_Package=Water_Hitting_WP

will display the variable as "Water Hitting WP" in most places in the application.

log.properties and logCriteria.properties

These files control which messages are logged and the name of the file they go to. The messages vary in severity from debug messages intended to help programmers maintain the software inlcuding "info" messages that provide information such as the names of files created by the graphical post-processor, "warn" messages that indicate the existence of conditions that in some contexts might indicate a behavior of the program that may not be the intended by the user, and "error" messages indicate the existence of a condition such as the program not being able to find a file. Upon termination, a message is written to the console indicating the location of the log file.

A fine degree of control is available for enabling and disabling messages.

outputfiles.properties

This file defines the graphical post-processor routines to be used for loading data from the TPA output files. This file would only be modified if new TPA variables or new parameters were added to the graphical post-processor.

variables.properties

This file is used to define the location of the variables and the plotting technique. It is not necessary to modify this file during common use of the graphical post-processor application.

2.2 Running the Program

Open an MS-DOS prompt console and apply the following instructions.

```
set GPP_HOME= c:\gpp\yy-mm-dd
```

Replace the previous path, c:\gpp\yy-mm-dd, by the appropriate home directory for the graphical post-processor.

```
set U_HOME=c:/Jsmith
```

Replace c:/Jsmith by the directory containing data from TPA runs or any other directory selected as the default directory. Note the use of /.

```
set MEMO=200m
```

Replace 200 by the desired allocated memory for the graphical post-processor application.

For Java™ to find the needed classes to run the graphical post-processor application, a variable known as classpath must be defined.

If the Java™ Runtime Environment was installed

```
set classpath=%GPP_HOME%
```

If the Java™ Development Kit was installed

```
set classpath=%GPP_HOME%;c:\jdk1.3\jre\lib\ext\visad.jar;c:\jdk1.3\jre\lib\ext\colt.jar
```

In the previous instruction it is assumed that the Java™ Development Kit was installed in the directory c:\jdk1.3. The user should replace c:\jdk1.3 by the appropriate directory containing the Java™ Development Kit.

To run the graphical post-processor, type

```
cd %GPP_HOME%
```

and

```
java –Xmx%MEMO% -Duser.home=%U_HOME% com.bayes.gpp.GPPApplet
```

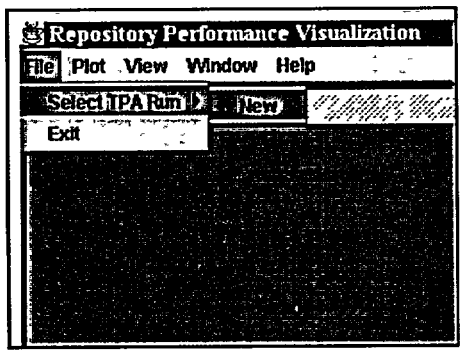
This last instruction will launch the main graphical post-processor window. It is recommended to write the previous MS-DOS instructions into a single batch file to make launching the graphical post-processor more efficient. In case of problems, open the file gpp\bayes.bat on the CD with a text editor and check the MS-DOS commands for launching the graphical post-processor. Alternatively, the batch file bayes.bat could be customized and executed from an MS-DOS prompt. The Repository Performance Visualization window will display on the screen after successful launching of the graphical post-processor.

2.2.1 Selecting TPA Data for Graphic Display

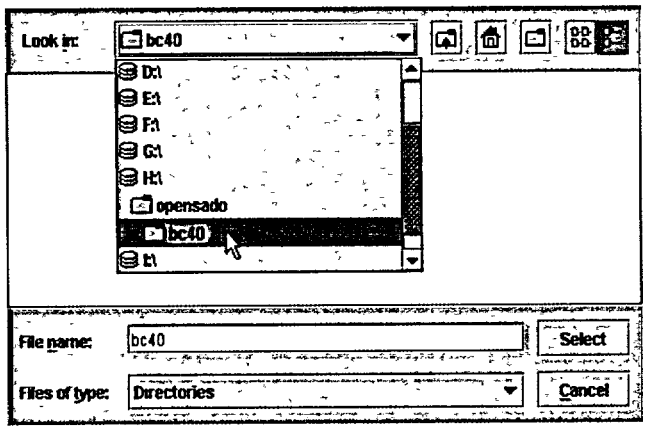
The Repository Performance Visualization window is a control window that allows identification of TPA the data to be displayed by the graphical post-processor. The TPA data must be located in a single directory. Appendix D lists all TPA output files required by the applications. Several of the needed output files are created only when the Append All Files flag is activated to run the TPA code.

Steps for the selection of the directory with the TPA data are

- Locate on the Repository Performance Visualization window and select File from the pull-down menu, then select TPA Run, then New.



- Identify the location of the folder for the TPA run to be used by clicking on the Look in pull-down menu.
- Select the drive where the run is located.
- Double click on the displayed folders to move from folder to folder.



- After locating the name of the folder where the run is located, ***single*** click on the folder, then click the Select button.

After selection of the TPA run, the Plot Control and View Control windows will be displayed, the latter is initially hidden behind the plot control window, which can be moved to a more convenient location on the screen.

2.2.2 Plot Control Window

The Plot Control window provides an intuitive interface for the selection of information to be displayed and the display mode. Particular elements of the Plot Control window are described as follows.

In the graphical post-processor application, a **variable** is defined as an element obtained as output of the TPA code. Most variables are functions of time, such as the dose rate. On the

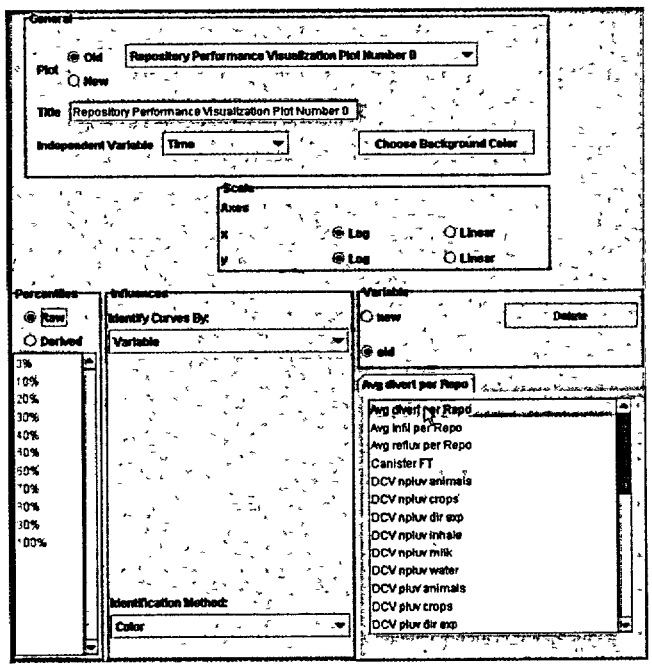
other hand, a **parameter** is defined as an element used as input to the TPA code, such as corrosion rate or sorption coefficient. For any parameter (e.g., the corrosion rate), the TPA code samples N values, where N is the number of realizations. A single value of the parameter is used for each of the TPA Monte Carlo realizations.

Some sample steps for the display of a variable are listed next.

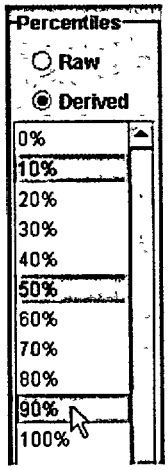
- Enter the name of a plot to be created. Replace the name in the Title field. If no title is written, the graphical post-processor will assign a default title of the form Repository Performance Visualization Plot Number N. This title will be shown as a header in the window containing the graphic display.

- Select log or linear scale, inside the Scale frame, using the radio buttons.

- Locate in the Variable frame and select a variable by making a single click on the variable name.



- Use the radio buttons on the left to select Raw or Derived for variables that are functions of time. If Derived is selected, identify the percentiles to be displayed. Use the Control key to select the percentiles individually or the Shift key to select a group (as in the Microsoft™ Windows operating system).



- Move to the View Control window and click on Apply. The selected variable will be displayed in the Repository Performance Visualization window.

The Plot Control window features a frame listing the variables that can be displayed by the graphical post-processor. In the Influences frame, a pull-down menu lists all TPA input parameters if Value of Parameter is selected or lists output variables if Value of Associated Variable is selected. The Influences frame is used to study the dependence between parameter and variable or between variable and variable. This feature is discussed in Sections 2.2.5 and 2.2.6.

To display variables, two choices are available in the Percentiles frame: Raw and Derived. When Raw data is selected, the plot shows curves for all realizations resembling a horse tail. When Derived is selected, the plot shows curves that do not intersect, corresponding to the percentiles selected. Examples of this feature are presented in Section 2.3.

Additionally, the Plot Control window includes a Scale frame for the selection of linear or logarithmic scales for the x and y axes. The title field is used to assign a title to a plot to be displayed in the Repository Performance Visualization window. The graphical post-processor assigns a default name for the plot if the user does not select one.

2.2.2.1 Sample Case of Graphic Display

This section includes instructions to display a sample plot to test the installation of the graphical post-processor. Test data generated with the TPA Code Version 4.0, 100 realizations of the non-disruptive base case, are included on the CD at a directory named TPAData

Steps for the display of a test plot are listed next.

- Locate the file gpp.properties at the directory $GPP_HOME. Open this file with a text editor and adjust it so that it contains

```
#which_subarea=2
num_realizations=100
param_percentiles=33,66,100
nuclide_of_interest=none
```

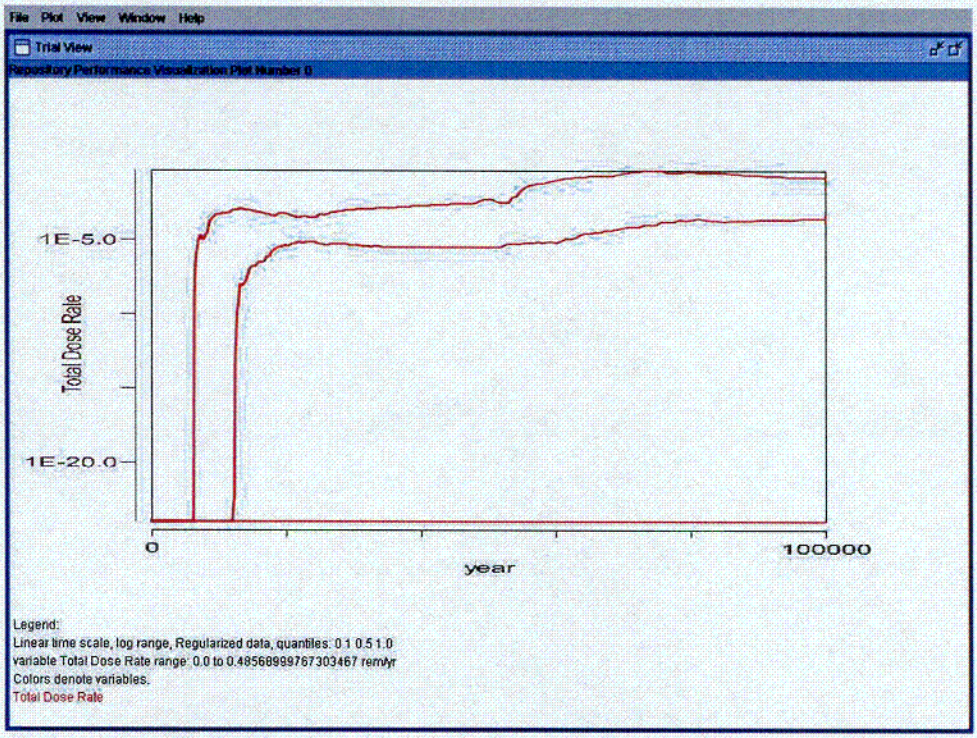
- Use the pound symbol, #, to comment lines out. Commenting out the line which_subarea causes the graphical post-processor to include all subareas, as discussed in Section 2.1.

- Launch the graphical post-processor as discussed in Section 2.2. It is important to first modify the file gpp.properties and then launch the graphical post-processor. Otherwise, changes to gpp.properties will not be recognized.

- Locate the directory TPAData on the CD. Select this directory from the file menu in the Repository Performance Visualization window as discussed in Section 2.2.1. The Plot Control and View Control windows will pop out after successful selection of the TPAData directory.

- Locate in the Plot Control window Variable frame and select Total Dose from the list of variables. In the Influences frame, make sure that Variable is selected. In the Percentiles frame, select the radio button for Derived. Select the 10th, 50th, and 100th percentiles, holding down the control key and right clicking with the mouse. In the Scale frame, select Linear scale for the x axis and Log scale for the y axis.

- Finally, locate the View Control window and hit the Apply button. The following plot must be displayed.



C01

2.2.3 View Control Window

A **view** is defined as a graphic in the Repository Performance Visualization window. The View

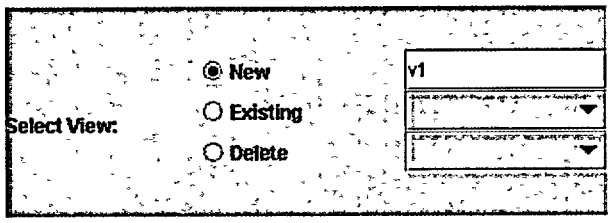
Control window is a panel that allows management of multiple views. Views can also be deleted or replaced using the View Control window.

2.2.3.1 Create or Add a New View

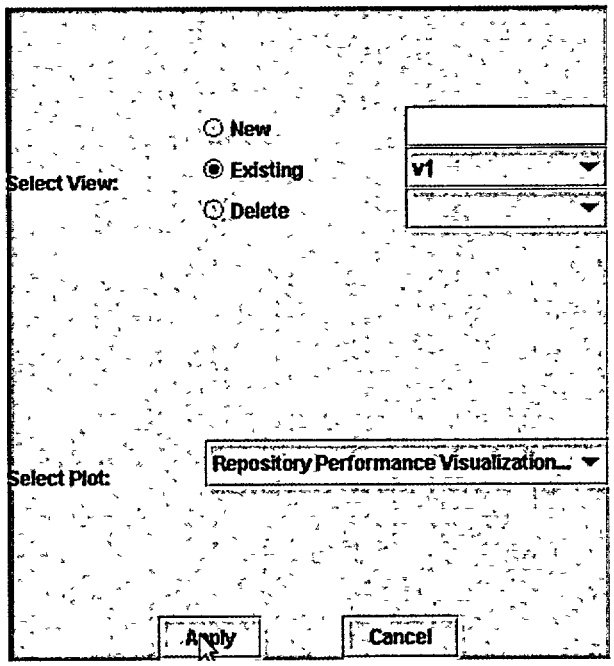
A collection of sample steps to create a new view is described as follows:

- Click on the New radio button next to the Select View label.

- Type a name of the new view, for example, v1.



- Go to Select Plot and select the name of the plot from the pull-down menu. Note that the name of the view now appears next to the Existing view label.

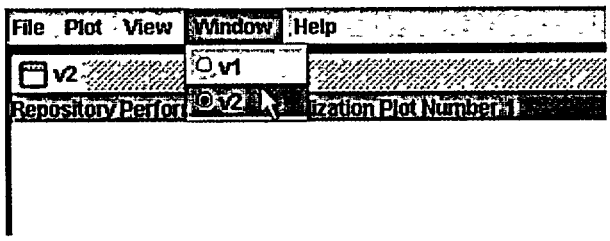


- Go to the Plot Control window and select the information to be displayed.

- Go back to the View Control window and Click the Apply button at the bottom of the window.

- To add another, view, such as v2, repeat the previous five steps replacing v1 with v2.

Note that the available views, v1 and v2 in the previous examples, appear in the Repository Performance Visualization window, below the Window menu. The corresponding view can be displayed in the Repository Performance Visualization window by clicking on the appropriate radio button in the Window menu.



2.2.3.2 Delete a View

The available views in the Window menu in the Repository Performance Visualization window can be deleted from the View Control window. To delete a view

- Locate the View Control window.
- Click on the Delete radio button.
- Select a view from the pull down menu next to the Existing line.
- Click on the Apply button at the bottom of the window.

2.2.3.3 Replace a View

To replace the data stored in a view, select the view next to the Existing line in the View Control window, select the data to be displayed in the Plot Control window, and click on Apply in the View Control window.

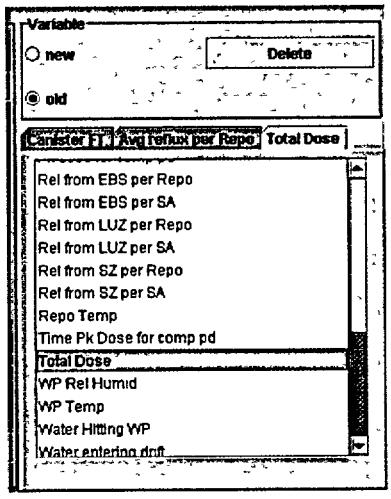
2.2.4 Displaying Several Variables on the Same Plot

The graphical post-processor provides the capability of displaying several variables on the same plot or view. It is recommended to display regularized or Derived data as opposed to Raw data to enhance the clarity of the plot. These are the steps to display multiple variables on the same plot.

- Locate the General frame in the Repository Performance Visualization window. Select the Old radio button next to Plot and choose the appropriate name of the plot from the pull-down list on the right.

- Locate the Variable frame in the Repository Performance Visualization window and click on the New radio button. A new field tab will be created. Single click on the variable name to be added to the plot.

- Repeat the above step for as many variables as desired to be displayed in the same plot. In the example in the next figure, three variables were selected.



- Make the desired selections in the Percentiles and Influences frames. Locate the View Control window and click on the Apply button. The plot will be displayed in the Repository Performance Visualization window. Note that additional labels are displayed on the vertical axis of the plot in the Repository Performance Visualization window. It is possible that some legends may fall outside the display window. If that is the case, the plot can be dragged to the right with the mouse to make all labels legible.

Delete a Variable from a Plot

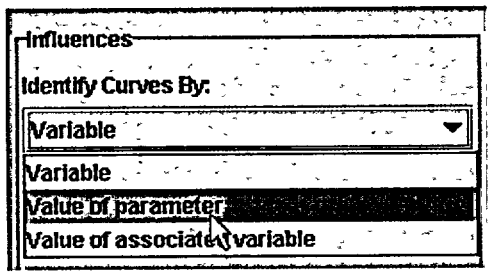
To delete a variable from a plot (there must be at least two variables on the plot to delete one), locate the Variable frame in the Plot Control window. Select the variable to delete, and click on the Delete button. The corresponding field tab will be deleted. The plot does not change automatically. To redraw the plot, make the appropriate selections in the Plot Control and View Control windows, and click on the Apply button in the view control window.

In the Plot Control window, in the Variable frame, field tabs with the label New Variable cannot be deleted. A variable must be selected, and its name displayed in the field tab, before it can be deleted.

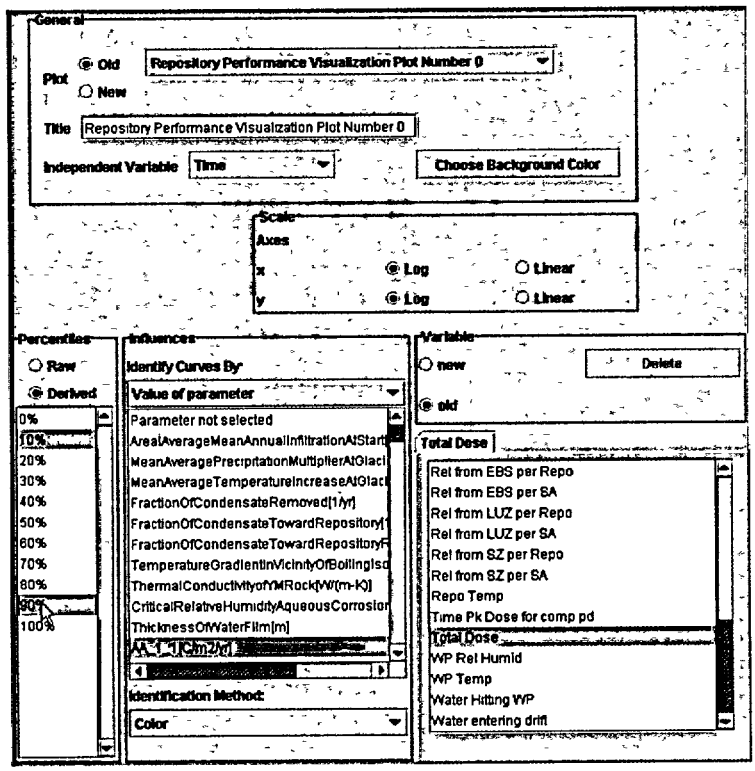
2.2.5 Influence of Parameters on Variables

The graphical post-processor provides some capability of studying the influence of parameters on variables (see Section 2.2.2 for definitions of parameter and variable). The steps to display an influence curve are as follows:

- Locate on the Plot Control window, inside the Influences frame, select the Identify Curves By: pull-down menu, and select Value of a Parameter.



- Make the appropriate selections in the Variable and Influences frames, selecting the variable to be displayed and the parameter influencing the output variable. Perform the desired selections in the Percentiles frame, selecting Raw or Derived data and the corresponding percentiles. Recall that multiple percentiles can be selected by holding the Control key and clicking with the mouse.

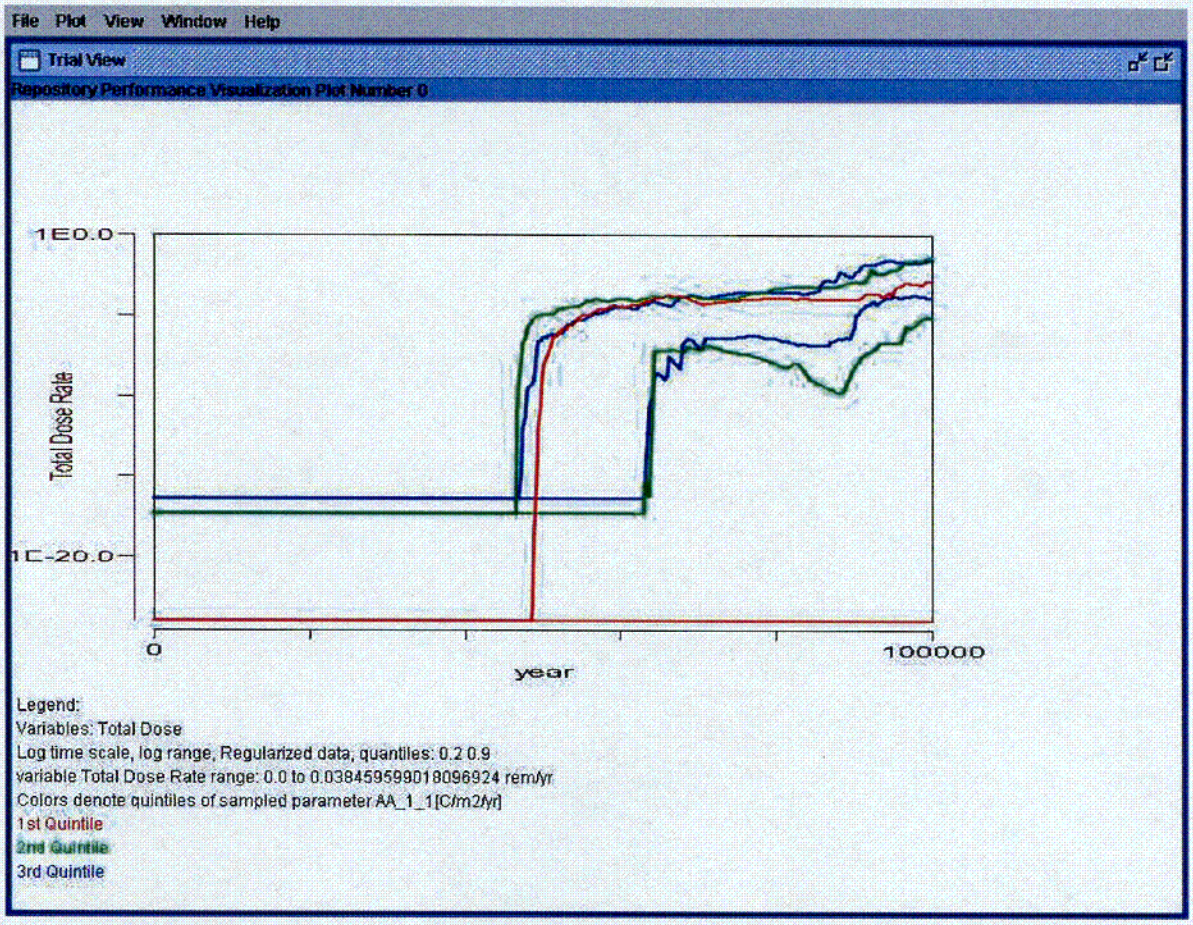


- Locate the View Control window, and click on the Apply button at the bottom of the window.

Figure 1 is used to explain the meaning of the influence diagram. This figure shows a sample plot, where the influence of the Alloy 22 corrosion rate (AA_1_1[C/m2/yr]) on the total dose rate is displayed.

In the TPA code, each sampled parameter has a value that, in general, differs from realization to realization. Let *g* represent a parameter, for example, the parameter AA_1_1[C/m2/yr] in Figure 1. For the case in Figure 1, the values of the parameter *g* are divided into three bins, based on their magnitude. Thus, assuming that a TPA run with 300 realizations is analyzed: the lowest 100 values of *g* are put into bin A, the next 100 go into bin B, and the last 100 values are classified into bin C. For each of the bins, the realization outputs (total dose rate for Figure 1) are drawn in the same color. Figure 1 only includes the 10th percentile and 90th percentile curves for each of the three bins.

The dose rates displayed in red correspond to those realizations with the lowest values of the corrosion rates. The dose rates in green are those realizations with intermediate values of corrosion rates. The doses in blue correspond to the highest corrosion rates. The trend observed is intuitively correct; the low doses are associated with those realizations with low corrosion rates and high doses to realizations with high corrosion rates. The way colors are assigned to the curves is explained in Appendix C.



C02

Note:

For any realization, multiple parameters are varied. The influence plots do not show the TPA output by varying a single parameter. Thus, influence plots must be interpreted with caution.

Influence plots are not available for box and whisker plots or for scatter plots.

The number of bins to be displayed in the influence plots is defined in the file gpp.properties and can be modified by the user with a text editor. A maximum of five bins can be displayed. For example, in file gpp.properties, the line

param_percentiles=33,66,100

indicates that the first bin encloses the 0 to 33rd percentile values of the parameter, the second bin ranges from the 33rd to 66th percentile, and the third bin encloses the 66th percentile to the maximum sampled value of the parameter.

In the following example, if the user defines

param_percentiles=10, 25, 50, 75, 100

five bins will be displayed in the influence plots. The first bin will range from the 0 to 10th percentile, the second from the 10th to 25th percentile, and so forth until the fifth bin, which will range from the the 75th to 100th percentile of the parameter values.

If the file gpp.properties is modified, the graphical post-processor application must be closed and opened again for the changes to be identified.

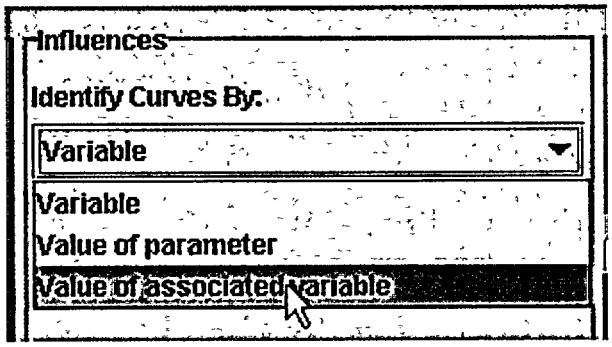
2.2.6 Influence of Variables on Variables

The graphical post-processor provides some capability for studying the influence of variables on variables (see Section 2.2.2 for a definition of variable). Consider, for example, variables A, representing the mean annual infiltration, and B, representing the flux of water available for radionuclide release. It is expected that realizations having high values of the mean annual infiltration also will have high values of the water flux contacting the waste package. The graphical post-processor can be used to display such relationships.

The technique used is similar to that described in Section 2.2.5. The main difference is that variables are functions of time. For each realization, values of the influencing variable (influencing and influenced variables are defined in the next paragraphs) are averaged for the time steps of the TPA run. Thus, there is a single average value of the influencing variable used in each realization. These averages are binned in the same manner as described in Section 2.2.5. For each of the bins, the influenced variable is drawn in a unique color. The technique to select the color is outlined in Appendix C.

The steps to display an influence curve are as follows:

- Locate on the Plot Control window, in the Influences frame, and select the Identify Curves By: pull-down menu. Select Value of Associated Parameter.



- Select a variable in the Influences frame. This is the **influencing** variable. In the Variable frame, select the variable to be displayed in the plot; this is the **influenced** variable. Make the desired selections in the Percentiles frame, selecting Raw or Derived data and the corresponding percentiles. Multiple percentiles can be selected holding the Control key and clicking with the mouse.

- Locate the View Control window, and click on the Apply button. Make sure the appropriate View to display the modified report has been selected in the View Control window.

NOTE:

The number of bins to be displayed in the influence plots is defined in the file gpp.properties and can be modified by the user with a text editor. See the notes in Section 2.2.5 for details.

Influence plots are not available for box and whisker plots or for scatter plots.

2.2.7 Scatter Plot for the Peak Dose

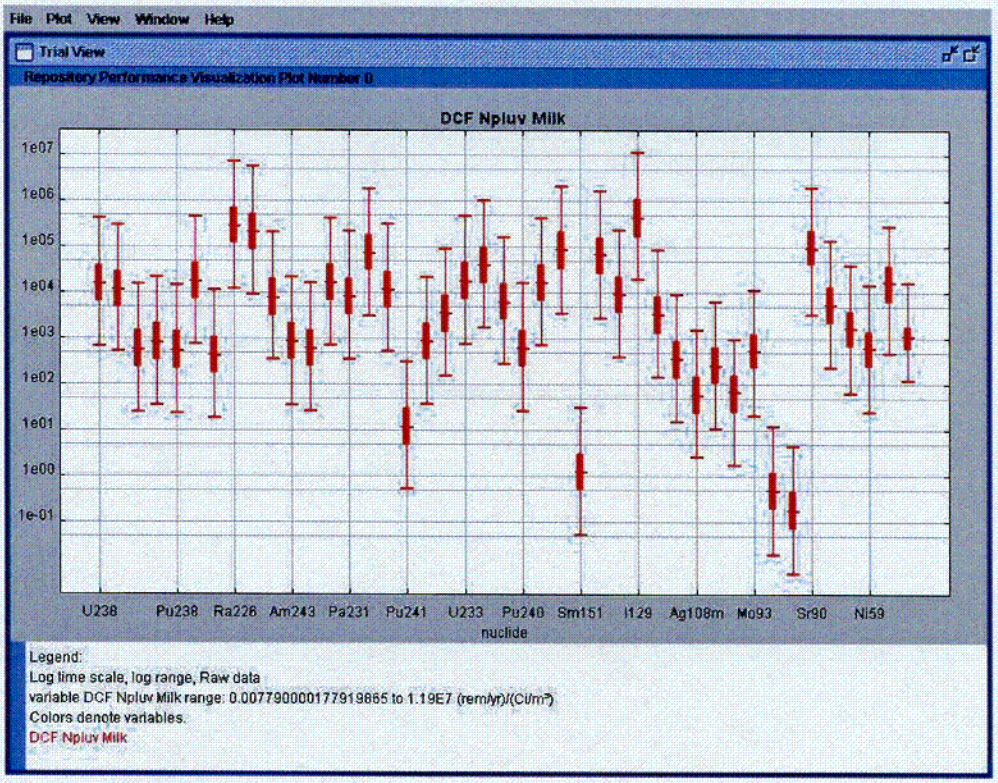
The graphical post-processor can display a scatter plot for the peak dose rate and the time at which the peak is reached.

In the Plot Control window, in the Variable frame, select the variable Peak Dose. The Influences frame becomes inactive (i.e., it is currently not possible to display influence scatter plots). A scatter plot is displayed after hitting Apply on the View Control window.

2.2.8 Box and Whisker Plots

Most of the variables are plotted versus time. However, dose conversion factors and radionuclide releases are plotted in the form of box and whiskers plots. These displays (dose conversion factors and radionuclide releases) include a box and whisker plot for each radionuclide. Each box and whisker plot summarizes relevant percentiles of the set of

considered values. The 20th to 80th percentile range is indicated by the extremes of the vertical rectangle. The median or 50th percentile is indicated by a horizontal line through the rectangle. The horizontal lines above and below the rectangle define the maximum and minimum of the set of values. Next is an example of a box and whisker plot.



NOTE:

The size of the graphic display does not allow listing of all of the nuclides in the plot. Only a few are listed. For TPA Version 4.0, box and whisker plots for dose conversion factors are listed in the order as they appear in the TPA file dcf.cum; i.e.,

1.	U238	16.	Cm245	31.	Sn121m
2.	Cm246	17.	Pu241	32.	Ag108m
3.	Pu242	18.	Am241	33.	Pd107
4.	Am242m	19.	Np237	34.	Tc99
5.	Pu238	20.	U233	35.	Mo93
6.	U234	21.	Th229	36.	Nb94
7.	Th230	22.	Cm244	37.	Zr93
8.	Ra226	23.	Pu240	38.	Sr90
9.	Pb210	24.	U236	39.	Se79
10.	Cm243	25.	U232	40.	Ni63

C03

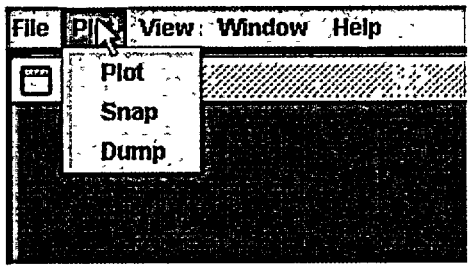
11.	Am243	26.	Sm151	41.	Ni59
12.	Pu239	27.	Cs137	42.	Cl36
13.	U235	28.	Cs135	43.	C14
14.	Pa231	29.	I129		
15.	Ac227	30.	Sn126		

For TPA Version 4.0, box and whisker plots for cumulative releases per nuclide are listed in the order as they appear in the TPA file cumrel.res; i.e.,

1.	Cm246	11.	Ra226
2.	U238	12.	Pb210
3.	Cm245	13.	Cs135
4.	Am241	14.	I129
5.	Np237	15.	Tc99
6.	Am243	16.	Ni59
7.	Pu239	17.	C14
8.	Pu240	18.	Se79
9.	U234	19.	Nb94
10.	Th230	20.	Cl36

2.2.9 Snap as a JPG or CSV File

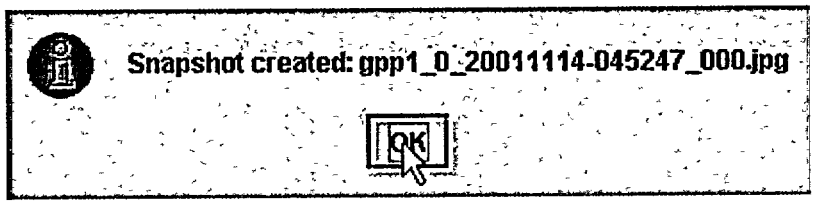
The graphical post-processor offers the possibility of saving the displayed data as a Joint Photographic Experts Group (JPG) file. Additionally, the data can be saved to a Comma-Separated Value (CSV) file that could be exported to Microsoft Excel. Exporting to Microsoft Excel is useful to have finer control on the graphic display.



To save a picture as a JPG file or CSV file

- Locate the Repository Performance Visualization window, and select Plot from the pull-down menu.

- Select Snap to snap the plot as a JPG image. The file will be saved and a message will display the name of the created file. The file will be located at the directory defined as $GPP_HOME. The file name contains information on the date and time at which the file was created.



- Select Dump to save a file as a CSV text file. The file will be saved and a message will display the name of the created file. The file will be located at $GPP_HOME. The file name contains information on the date and time at which the file was created.

NOTE:

Information in the CSV files is organized by rows. The elements of the domain (i.e., the time steps) are indicated by the word time at the beginning of the row. Subsequent rows represent values of the abscissa (i.e., values of the variable). In the case of influence plots, the time steps are repeated in multiple rows to differentiate the lines associated to multiple bins.

The CSV file creation option is not supported for box and whisker plots or for scatter plots.

Microsoft Excel is limited to handling a maximum of 256 columns. If the number of time steps is greater than 256, CSV files will be truncated by Excel to display only the first 256 time steps. In general, CSV files will be well displayed by common text editors.

2.3 Sample Plots

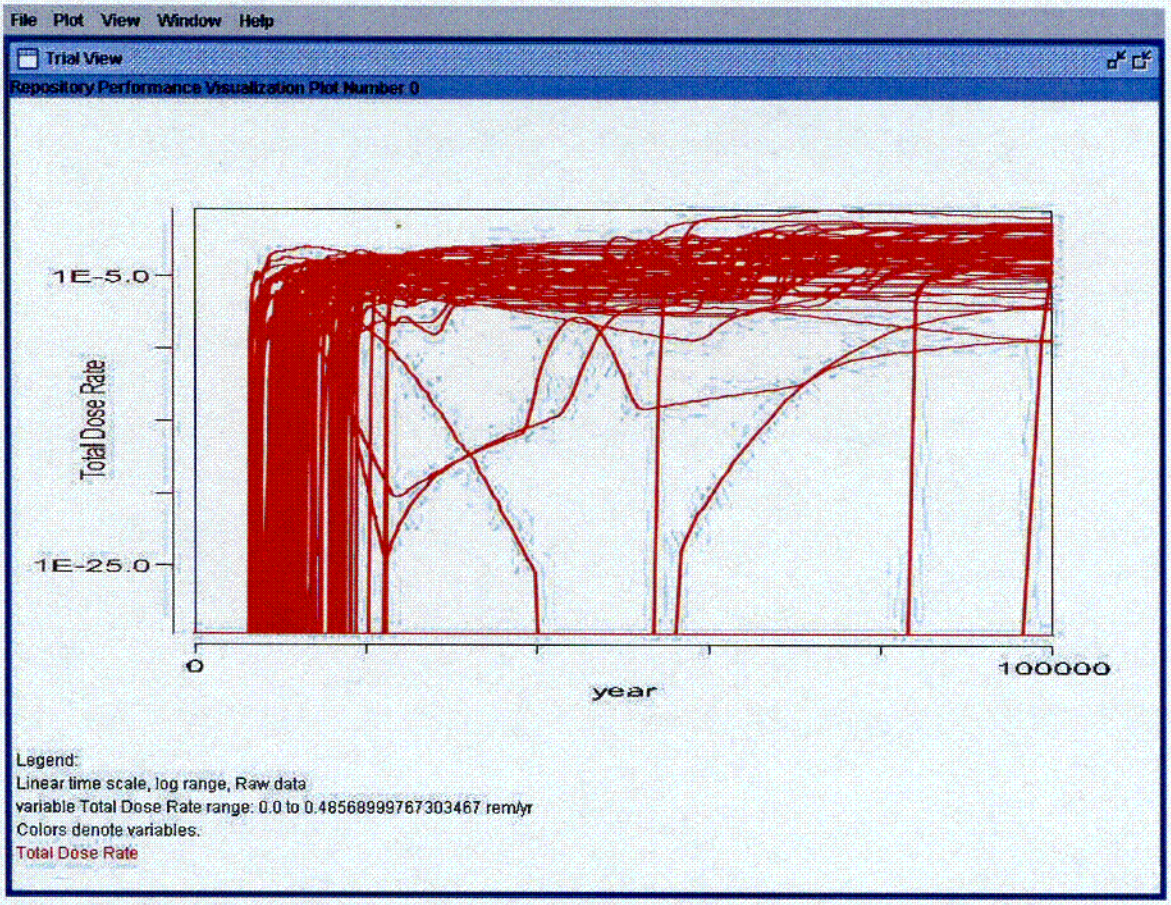
The following plots are representative graphic displays obtained with the graphical post-processor for the TPA code Version 4.0. All of these plots were generated with the sample data on the CD located in the directory titled TPAData. The file gpp.properties was modified with a text editor as

```
which_subarea=3
num_realizations=100
```

param_percentiles=33,66,100
nuclide_of_interest=none

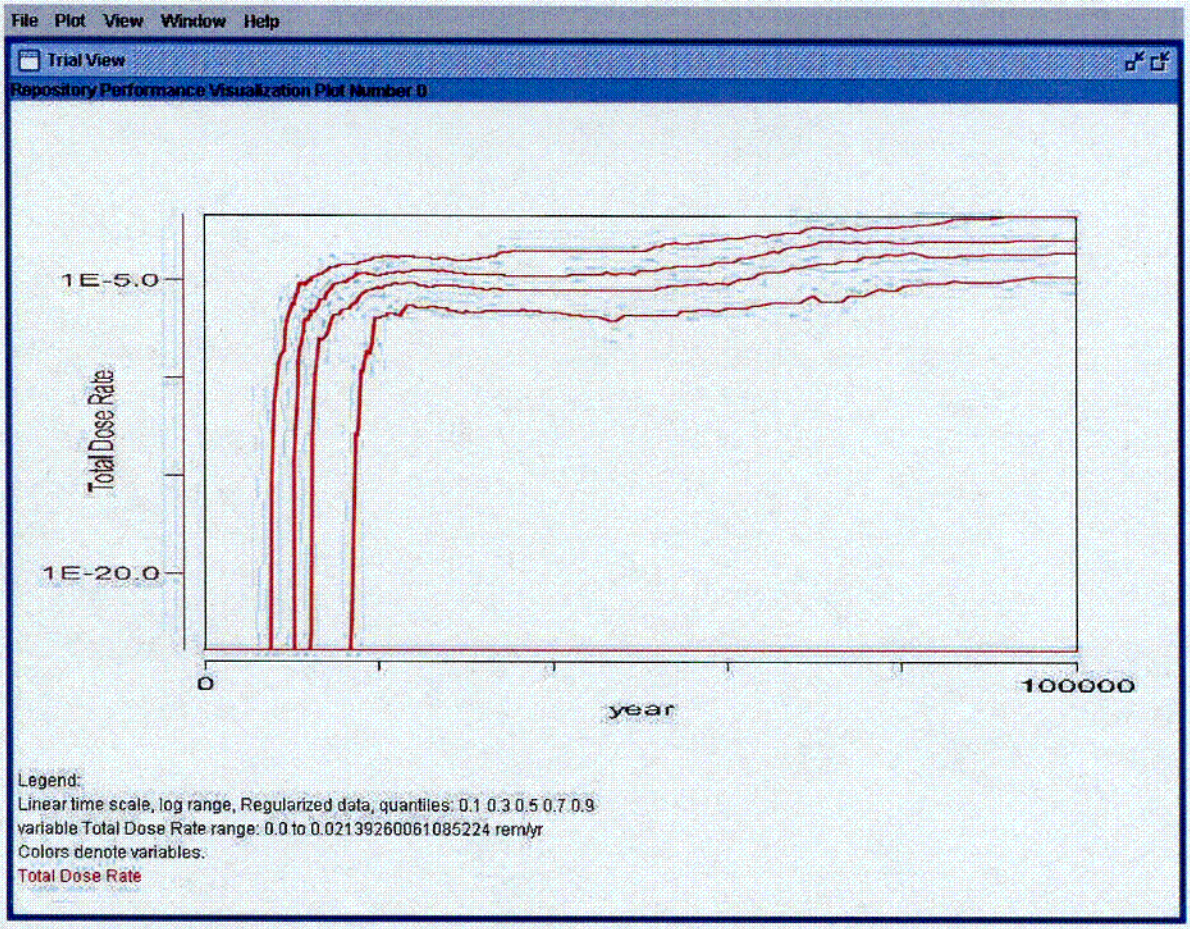
Recall that the graphical post-processor must be launched only after modification to the property files. Sample graphic displays are next presented. They could be used to test the installation of the graphical post-processor.

Display of raw data for total dose rate:



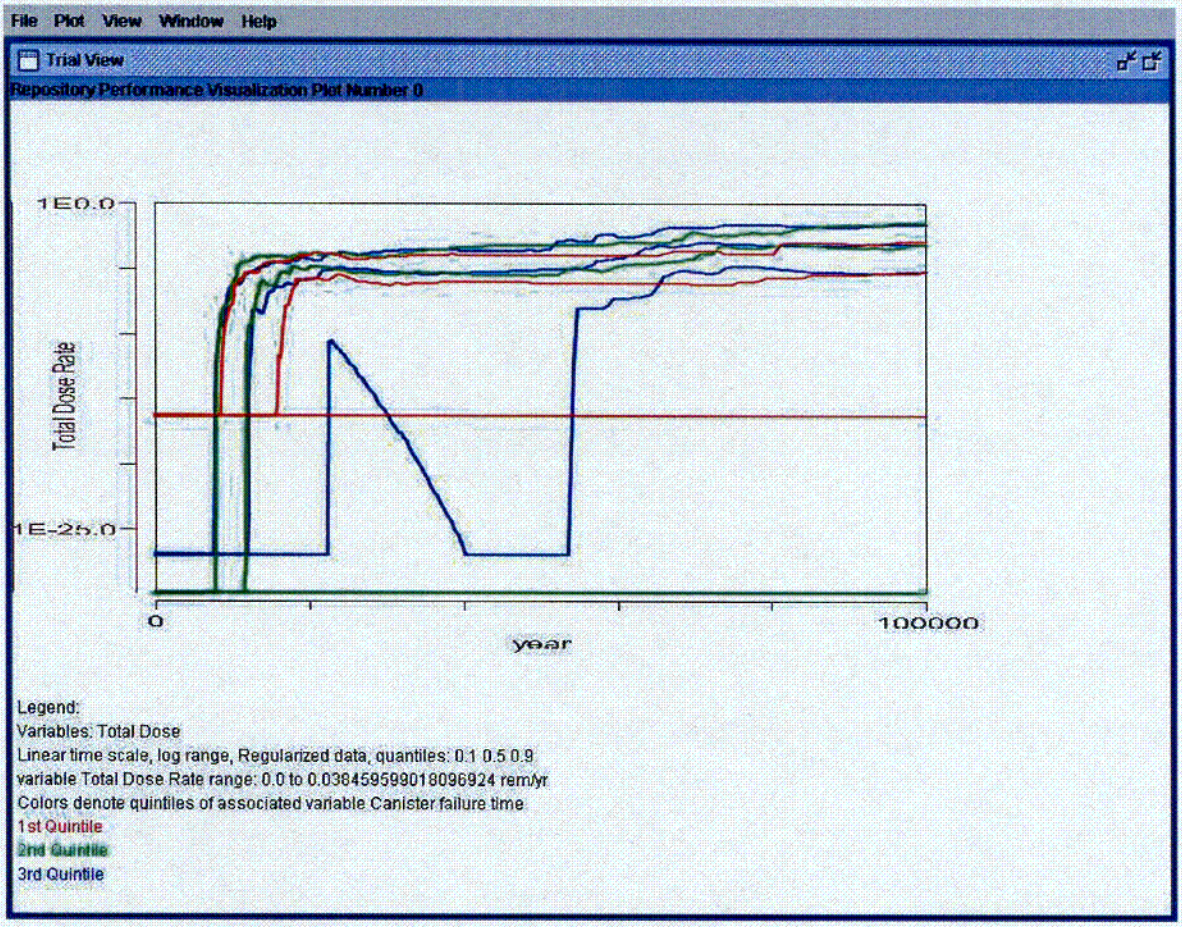
C04

Dose to receptor using derived (regularized) data for the 10th, 30th, 50th, 70th, and 90th percentiles:



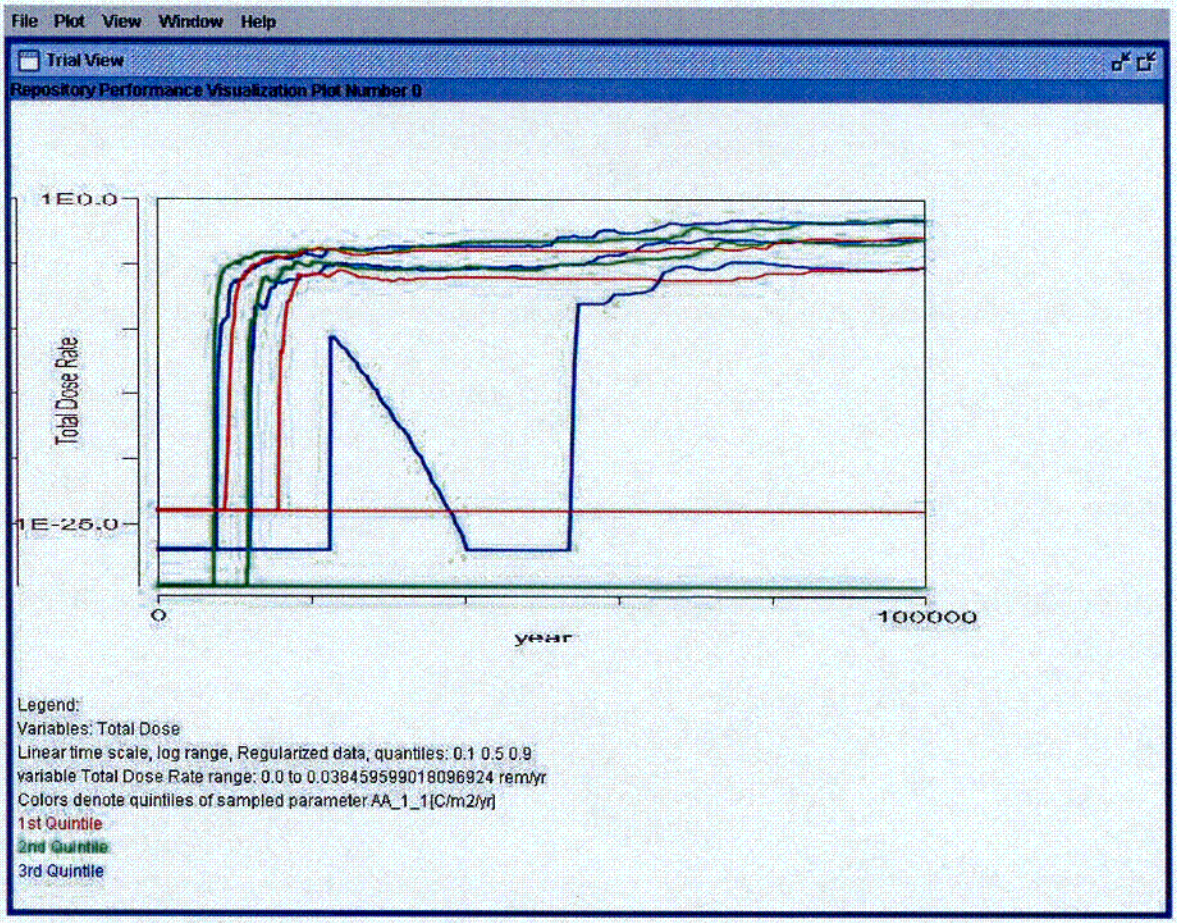
C05

Influence of Fraction of Waste Packages Failed on the dose rate. The red doses are associated with those realizations with few waste packages failed. Blue doses correspond to realizations with a large number of waste packages failed. Green doses belong to realizations with an intermediate number of waste packages failed. For each of the three bins (i.e., the red, green, and blue bins), three percentiles are displayed, 10th, 50th, and 90th percentiles.



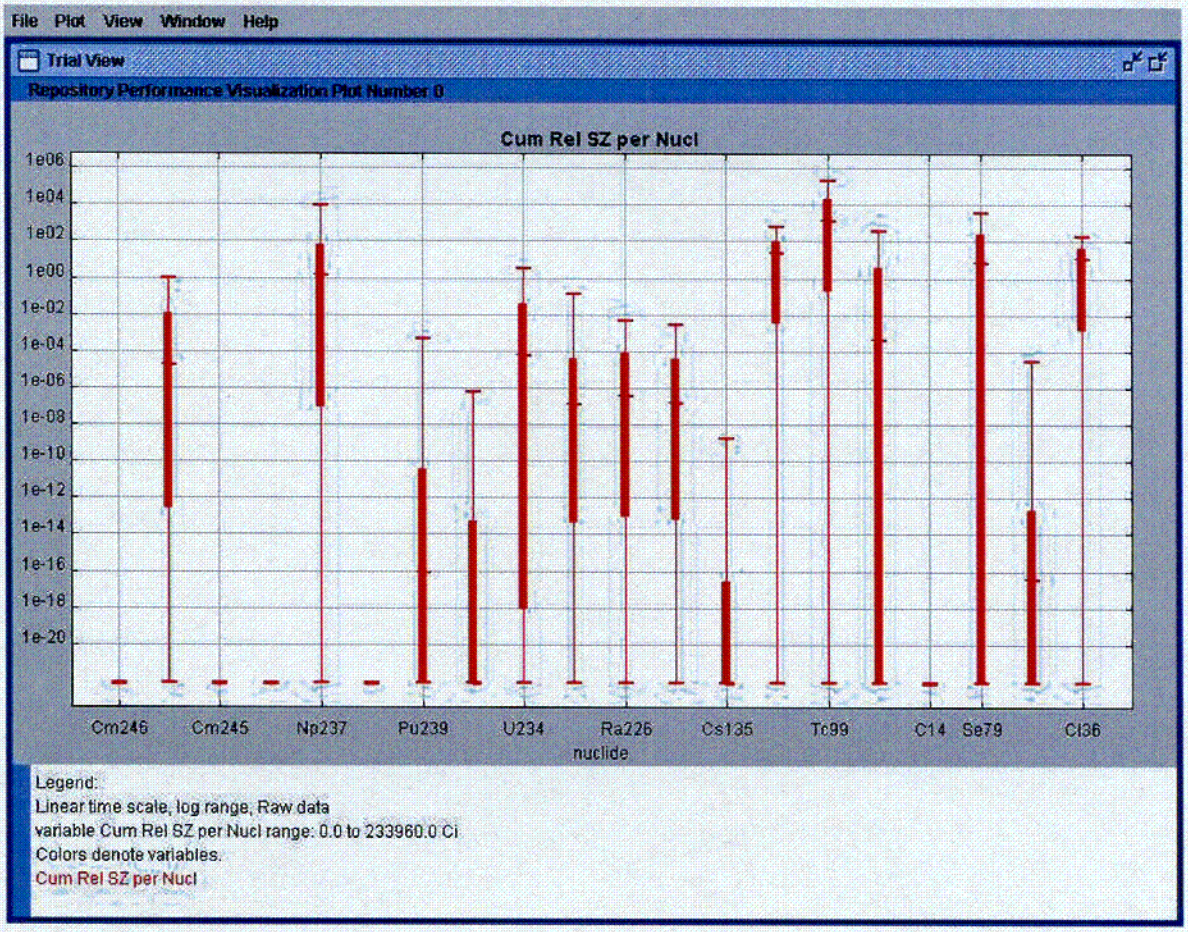
C06

Influence of Alloy 22 corrosion rates (AA_1_1[C/m2/yr]) on the total dose rate. Red, green, and blue doses correspond to realizations with low, intermediate, and high corrosion rates, respectively. For each of the red, green, and blue bins, three percentiles are plotted (10th, 50th, and 90th percentiles).



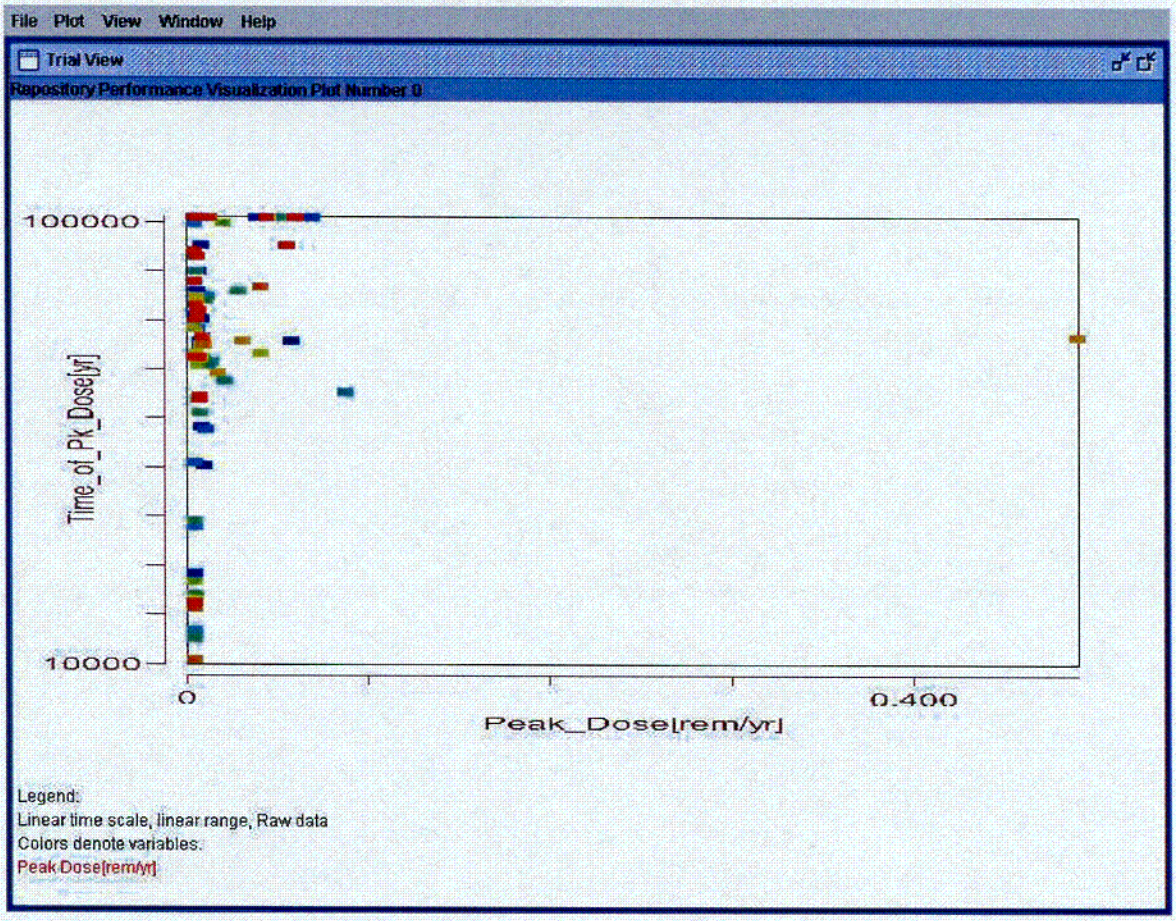
C07

Release from lower unsaturated zone per radionuclide.



C08

Scatter plot of the time of the peak dose versus the peak dose for each realization.



NOTE:

This release of the graphical post processor does not allow control of the scales for the x and y axes. Both axes are drawn in linear scale.

C09

APPENDIX A

OBTAINING THE JAVA™ FROM THE INTERNET

The Java™ 2 Platform, Standard Edition, is freely available at

http://www.javasoft.com/products/index.html

Microsoft™ Windows versions of the Java™ Development Kit or Java™ Runtime Environment are available at the above site.

For other operating systems, consult the following sites.

Java™ Development Kit 1.2.2 for Solaris:
http://java.sun.com/products/jdk/1.2/

Java™ Development Kit 1.1.8 for Solaris:
http://java.sun.com/products/JDK/1.1/

Java3D™ 1.1.3 for Solaris SPARC:
http://java.sun.com/products/java-media/3D/index.html

Windows 95/98/NT
Java™ Development Kit 1.2.2 for Windows (SUN):
http://java.sun.com/products/jdk/1.2/

AIX
Java™ Development Kit 1.2.2 for AIX:
http://www.ibm.com/java/jdk/download/index.html

Java™ Development Kit 1.1.8 for AIX:
http://www.ibm.com/java/jdk/download/index.html

HPUX
Java™ Development Kit 1.2.2:
http://www.unixsolutions.hp.com/products/java/hpux11_releases.html

Java3D™ 1.1.3:
http://www.unixsolutions.hp.com/products/java/hpux11_releases.html

IRIX
Java™ Development Kit 1.2.X for IRIX:
http://www.sgi.com/developers/devtools/languages/java.html

Java3D™ 1.1.3 for IRIX:
http://www.sgi.com/developers/devtools/languages/java.html

True 64 Unix (formerly Digital Unix)
Java™ Development Kit 1.2.X:
http://www.digital.com/java/download/index.html

Java™ Development Kit 1.1.8:
http://www.digital.com/java/download/index.html

Linux
Java™ Development Kit 1.2.X (from Blackdown.org):
http://www.blackdown.org/java-linux/jdk1.2-status/

Java™ Development Kit 1.2.X (from SUN):
http://java.sun.com/products/jdk/1.2/

Java™ Development Kit 1.1.7 (Blackdown):
http://www.blackdown.org/java-linux.html

Java3D™ 1.1.3 (Blackdown):
http://www.blackdown.org/java-linux.html

APPENDIX B

SOFTWARE MAINTENANCE

A criterion for selecting the technical approach to implement the graphical post-processor application was minimizing the cost and difficulty of software maintenance.

The application consists of Java™ source files, their compiled images, and a set of properties text files that describe the structure of the Total-system Performance Assessment (TPA) files to be analyzed. Each source file performs the following tasks:

- Extract data from TPA output files
- Process data
- Plot results of computations
- Manage graphical user interface

To extend the graphical post-processor to process additional variables, new entries must be made to the properties files, to

- Identify the new variable, the source file, and the location within the file
- Identify the new file, provided it is not already identified by the data in use by the graphical post-processor

In the event that the file with the new variable is organized in a different manner from the files currently in use, a new load strategy consistent with existing load strategies must be added.

If line numbers are not contained in those messages, the maintainer can force line numbers to be dislayed by executing the software with the Java™ command argument-DJAVA_COMPILER=NONE. In some cases, recompiling the software with debug enabled may produce useful information.

If a bug is found in the software, typically the Java™ runtime emits a trace indicating the line of the code that could not be executed, and the stack of procedures from which it was called. The Java™ runtime responds to a bug by emitting a trace pointing at the line of the code that could not be executed, and the stack of procedures from which it was called.

APPENDIX C

STRATEGY FOR THE SELECTION OF COLOR

When there is a need to distinguish multiple curves on the same plot, the graphical post-processor assigns each curve and its associated descriptive text a different style. In the initial release of the graphical post-processor, style means color, therefore, each curve is assigned a distinct color. The colors used are fully saturated (i.e., as close to single-wavelength as the hardware supports), with maximum luminance. Each member of a sequence of curves is assigned a unique hue. The colors assigned to successive members of the sequence range from red to violet and are approximately equidistant, in the sense of being separated by the same number of just-noticable differences of color.

APPENDIX D

TOTAL-SYSTEM PERFORMANCE ASSESSMENT (TPA) FILES

The collection of Total-system Performance Assessment (TPA) files used by the graphical post-processor is listed in the following table. The names in the column on the left are dummy names, used by the graphical post-processor in its internal routines. The TPA files used by the graphical post-processor can be specified in the two property files: variables.properties and outputfiles.properties. Modifications in these property files must be consistent, otherwise the graphical post-processor application may not perform correctly.

The display name for the variables in the Plot Control window is defined in the property file variable_abbrevs.properties. These display names are also used in the graphical post-processor plots.

Graphical Post-processor Variable	TPA Output File
Repository_Temperature	nfenv.rlt
Waste_Package_Temperature	nfenv.rlt
Waste_Package_Relative_Humidity	nfenv.rlt
Water_Hitting_Waste_Package	nfenv.rlt
Total_Dose	totdose.res
Infiltration_per_subarea	uzflow.rlt
Canister_failure_time	ebsfail.rlt
Water_entering_drift	ebsflo.dat
Release_from_saturated_zone_per_subarea	szft rlt
Release_from_lower_unsaturated_zone_per_subarea	uzft.rlt
Release_from_engineered_barrier_system_per_subarea	ebsrel.rlt
Mean_annual_temperature	dcagw.ech
Mean_annual_precipitation	dcagw.ech
Release_from_engineered_barrier_system_per_repository	cumrel.res
Release_from_lower_unsaturated_zone_per_repository	cumrel.res
Release_from_saturated_zone_per_repository	cumrel.res
Peak_dose_for_compliance_period	gwpkds_c.res
Average_infiltration_per_repository	infilper.res
Average_reflux_per_repository	infilper.res
Average_diversion_per_repository	infilper.res
DCF_pluvial_direct_exposure	dcf.cum
DCF_pluvial_inhalation	dcf.cum
DCF_pluvial_ingestion_animal_prod	dcf.cum
DCF_pluvial_ingestion_crops	dcf.cum
DCF_pluvial_drinking_water	dcf.cum
DCF_pluvial_milk	dcf.cum
DCF_nonpluvial_direct_exposure	dcf.cum
DCF_nonpluvial_inhalation	dcf.cum
DCF_nonpluvial_ingestion_animal_prod	dcf.cum
DCF_nonpluvial_ingestion_crops	dcf.cum
DCF_nonpluvial_drinking_water	dcf.cum
DCF_nonpluvial_milk	dcf.cum
Time_of_peak_dose_for_compliance_period	gwpkdos.res

Properties of the TPA run and names of the parameters are read from the TPA files tpa.inp and samplpar.hdr. Sampled values of the parameters are obtained from the file samplpar.res.