SAPHIRE Basics

An Introduction to Probabilistic Risk Assessment via the Systems Analysis Program for Hands-On Integrated Reliability Evaluations (SAPHIRE) Software

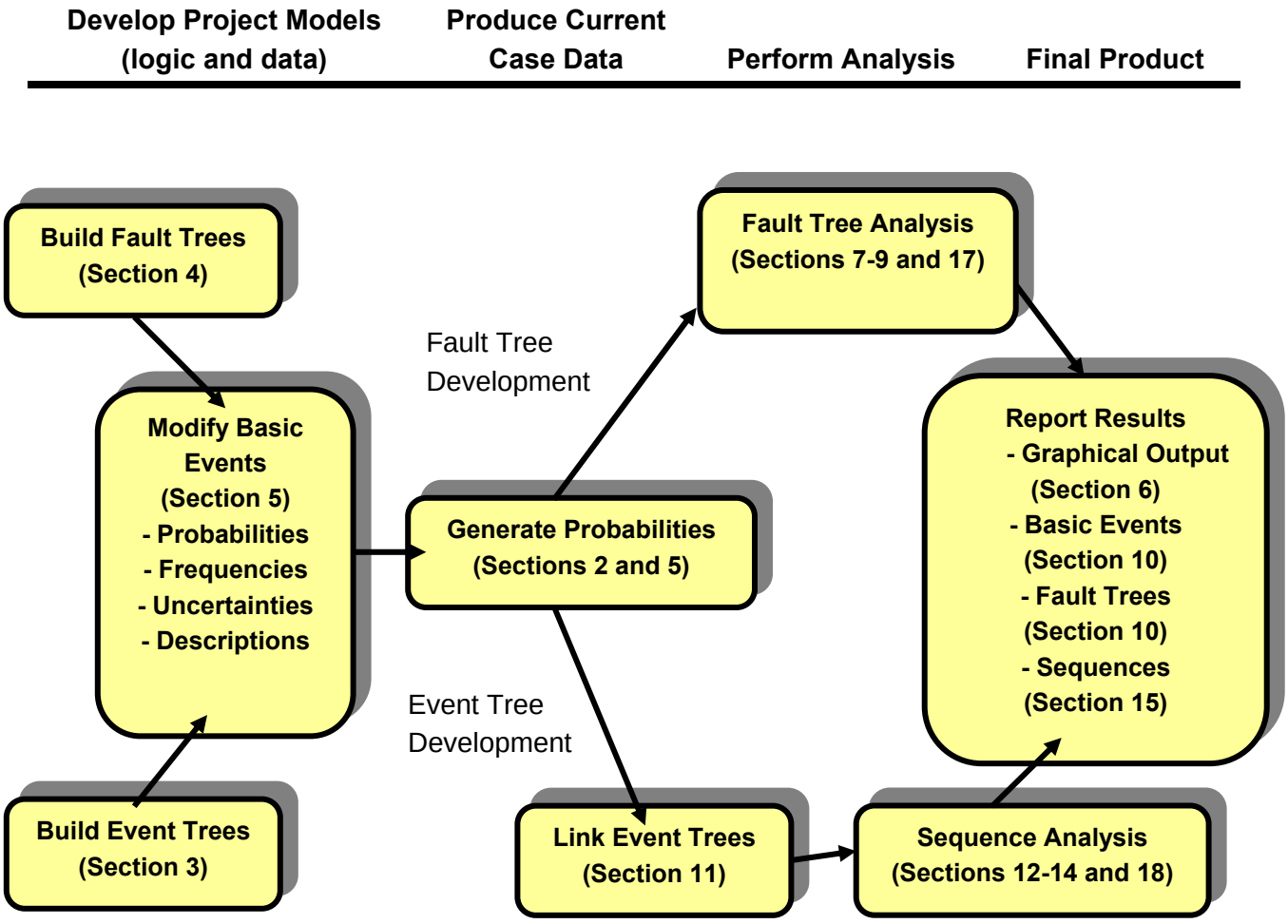
Curtis Smith
James Knudsen
Michael Calley
Scott Beck
Kellie Kvarfordt
Ted Wood

Idaho National Laboratory

January 2009

Idaho National Laboratory

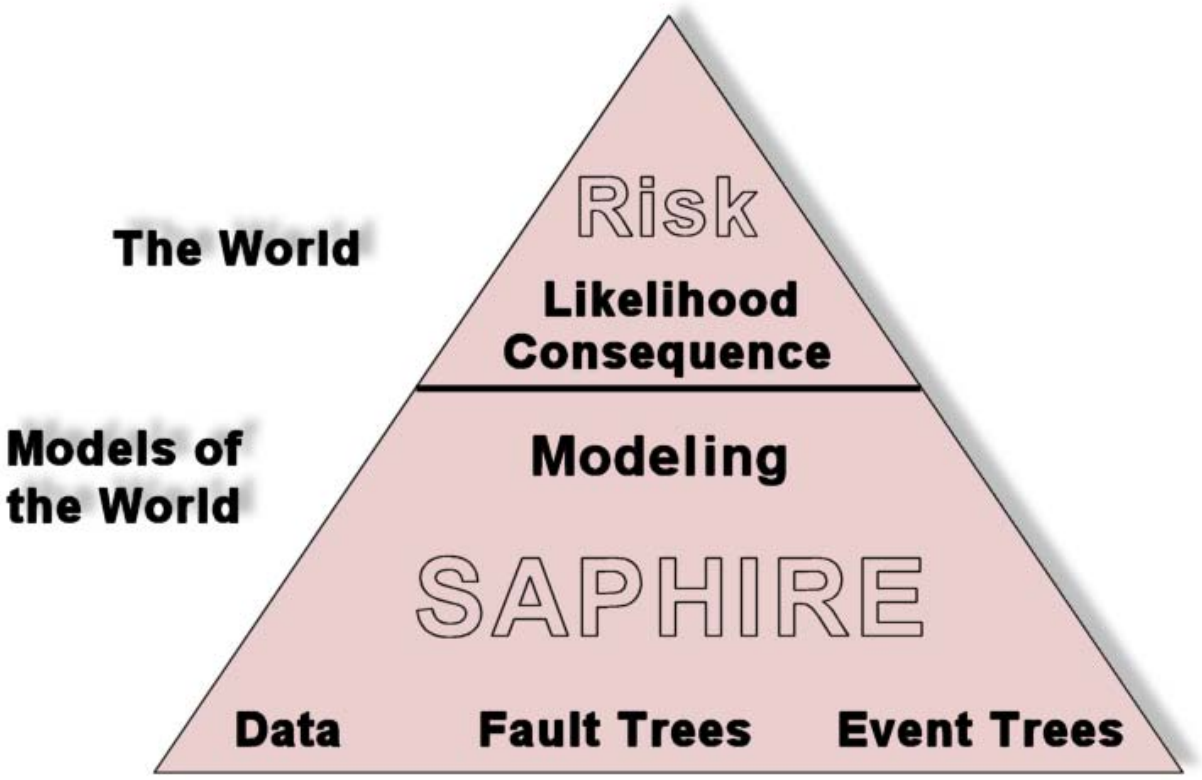
SAPHIRE – The "Big Picture"



CONTENTS

1 INTRODUCTION

Section 1 contains an introduction to the SAPHIRE Basics course material, an overview of PRA, and important definitions and concepts.



1.1. Overview of SAPHIRE Basics Course Material

The SAPHIRE Basics course material is intended to both (1) provide guidance for learning SAPHIRE during the Basics class and (2) become a stand-alone reference document after finishing the class. Thus, the format for the class material is a combination of the traditional "overhead-type" of presentation information with a structured, reference-type document.

SAPHIRE screen displays will be shown as they appear on your video display (as shown below).

When discussing a particular sequence of menu options, the nomenclature

MAIN Menu ➔ Submenu Option

will be used to indicate the main SAPHIRE menu option and any successive submenu options.

1.2. SAPHIRE - What Is It and What Can It Do?

- SAPHIRE is an integrated PRA software tool that gives a user the ability to create and analyze fault trees and event trees using a personal computer.

- IRRAS was originally released in 1987 (version 1.0). Other versions of IRRAS include 2.0, 2.5, and 4.0. Additions and improvements have been added to each version of the code.

- Creation of 32-bit IRRAS, version 5.0, in 1992 resulted in an order of magnitude decrease in analysis time. New features included: individual codes modules combined into a single module; end state analysis; fire, flood, and seismic modules; rule-based cut set processing; and rule-based fault tree to event tree linking.

- SAPHIRE for Windows, version 6.0, is released in 1997. Use of a Windows user interface makes SAPHIRE easier to learn and use.

- SAPHIRE for Windows, version 7.0, is released in 1999. This is latest version of the SAPHIRE code. This manual is written for version 7.x of the software.

- SAPHIRE contains several features:

    - PC-based fault tree and event tree graphical and text editors
    - Cut set generation and quantification
    - Importance measures and uncertainty modules
    - Relational database with cross-referencing features
    - External events analysis (e.g., seismic, location transformation)
    - Rule-based recovery and end-state analysis
    - Common Cause Failure (CCF) basic event capabilities

- SAPHIRE minimal hardware requirements:

    - Windows 98 or greater
    - Pentium class IBM-PC compatible with 2-button mouse
    - 50 MB free disk space (minimum for installation)

1.3. The Class Workbook

- The workshop problems for the SAPHIRE class are contained in a separate handout, referred to as the “workbook” or "workshop manual."

- The workbook allows the SAPHIRE Basics class to be tailored to specific audiences. This “tailored-problem” format gives the freedom to present specific topics or problems centered around the expected needs of the students.

- The workbook follows the same format as the course material, and together they provide an integrated reference package for the SAPHIRE code.

1.4. Overview of PRA

- Probabilistic risk assessment (PRA) is a method to:

Identify, characterize, quantify probabilistically, and evaluate *hazards*

- The process of measuring risk (i.e., PRA) asks:

  - What can go wrong?
  - How likely is it?
  - What are the consequences?

- Hazards could include
  - Ionizing radiation (e.g., a nuclear power plant radiation release)
  - Electrical hazard (e.g., electrical shock)
  - Thermal hazard (e.g., thermal blast effects from an explosion)
  - Chemical hazard (e.g., a release of toxic chemicals)

1.5. Definitions

♦ Risk

The potential of loss or damage resulting from exposure to a hazard.

♦ Safety

Represents an acceptable level of risk relative to the benefits derived from the hazards-causing activity.

♦ Probability

The two common interpretations of *probability* are:

Frequentist *(the relative frequency or empirical approach)* — The probability of event A is given by:

$$P(A) = \lim_{n \to \infty} (x/n)$$

where *X* is the number of times event A occurred out of *n* number of repeated trials. For a fixed *n*, the value of P(A) is the *relative frequency* of occurrence of event A. Consequently, increasing *n* will improve the estimate of P(A).

Subjective *(the "degree of belief" approach)* — The probability P(A) is the measure of uncertainty or degree of belief one has of event A. For example, the knowledge of symmetry for a particular coin may lead an analyst to postulate that the probability of tossing a head on a toss is 0.5. The subjective method requires that probability be assigned in a consistent manner.

♦ Reliability

The probability that a system will perform satisfactorily (i.e., does not fail) for a designated period of time (or number of cycles) and under specified operating conditions. The **Unreliability** is the complement of the reliability, that is, the probability that the system *does fail* within a designated period of time and under specified operating conditions.

- Availability

The instantaneous availability is the probability that a system will perform satisfactorily at a designated point in time when used under specified operating conditions. The evaluation of system availability includes operating time, time to test, active repair time, administrative time, and logistics time. The **Unavailability** is the complement of the availability.

- Accident Sequence

The combination of an initiating event with system failures and successes (defined by an event tree) which results in a definable outcome. For a nuclear power plant PRA, the outcome is generally core damage.

- Dominant Contributors

Failures which are quantitatively the largest contributors (i.e., “dominant”) to the likelihood of the defined event (e.g., accident sequence, system failure).

- Minimal Cut Set

A minimum combination of failures needed to result in the occurrence of the event of interest (e.g., accident sequence, system failure).

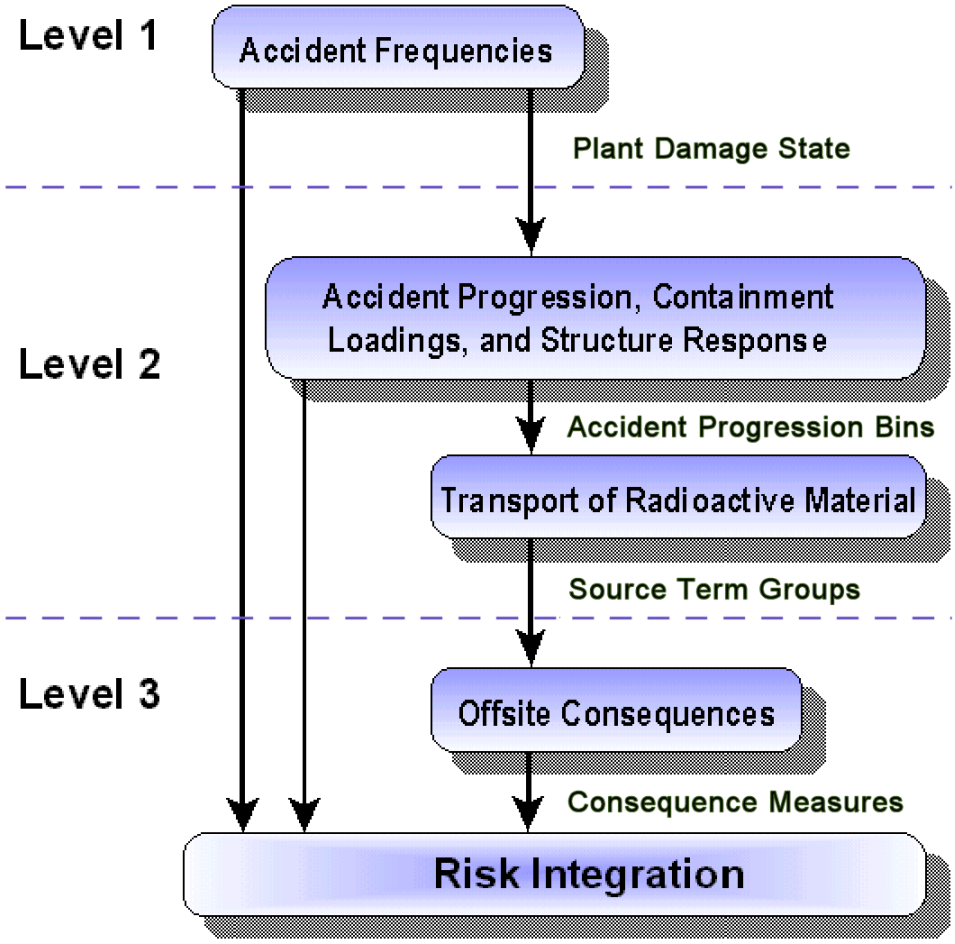
- Consequence

A measure of the degree of damage or loss experienced given a particular accident sequence. For a nuclear power plant PRA, an accident sequence consequence is generally the anticipated offsite radiation dose to the population surrounding the plant.

- Fault Tree Linking

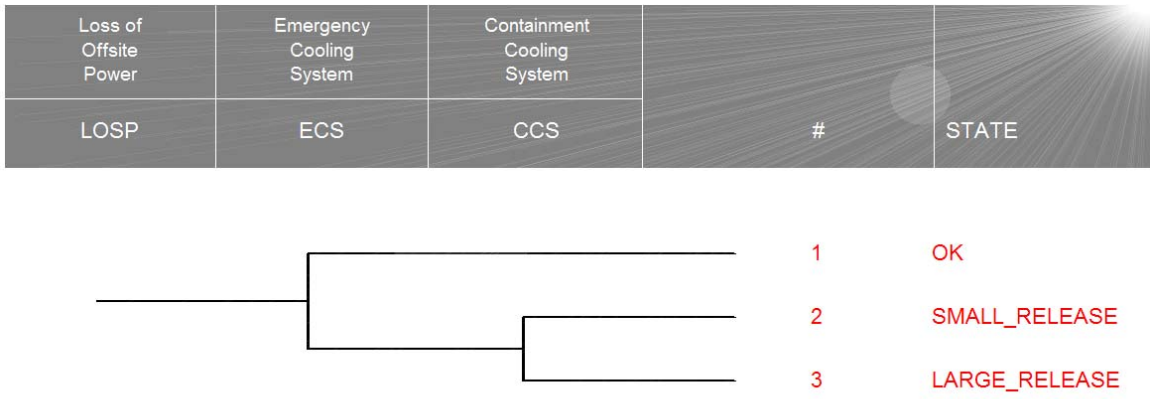
A technique whereby the fault tree logic is combined with the event tree logic (i.e., successes and failures) resulting in a logic expression for each sequence in the event tree.

1.6. Major Steps "NUREG-1150 Type" PRA Process



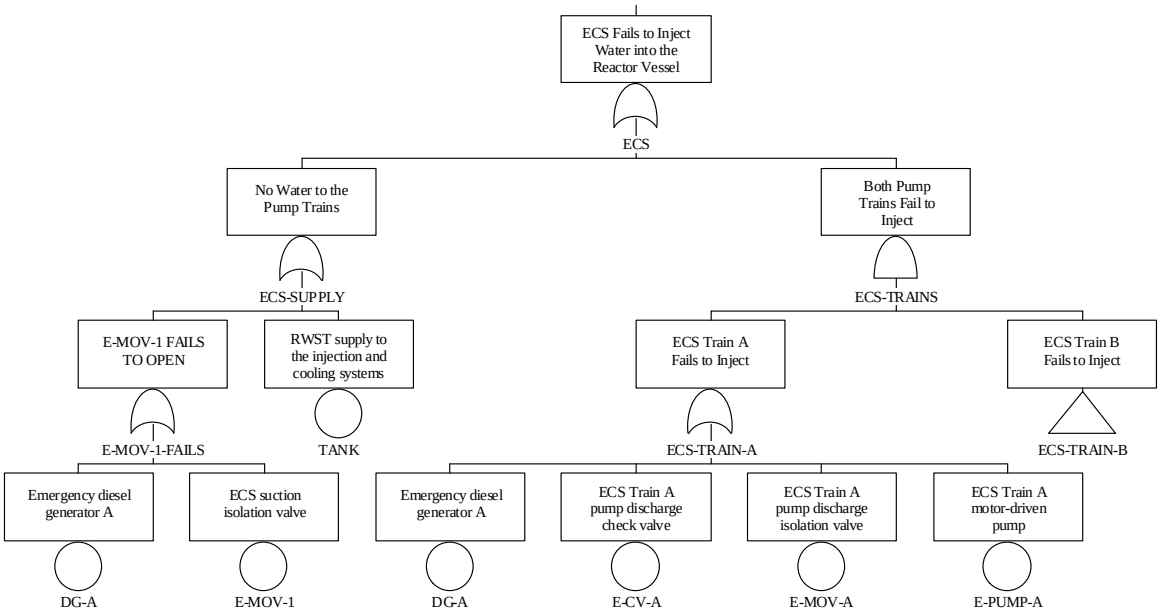
- The output of the Level 1 PRA is the core damage frequency and includes:.
  - ◊ Identification of accident sequences and their frequencies.
  - ◊ Identification of dominant contributors to core damage.
  - ◊ Classification of accident sequences into Plant Damage States.
- Event tree and fault tree analysis are most commonly used in Level 1 PRA.

Event Trees



- Event trees are logical representations of significant plant responses to initiating events.
  - Each sequence results in either a safe condition (e.g., safe shutdown) or an accident condition (e.g., core damage).
  - Event trees relate systems/functions to a sequence progression.
  - Event trees provide an end-to-end traceability of accident sequences.
- Event trees provide a traceable way to perform the following functions:
  - Identify accident sequences.
  - Identify essential safety system functions.
  - Quantify sequence frequencies.

Fault Trees



- Fault trees are logical representations of the credible failures that can cause an undesired event to occur.
  - The undesired event is stated at the top of the fault tree.
  - The fault tree gates specify the logical combinations of basic events that lead to the top event.
  - Fault trees can be used to identify system “weaknesses.”
  - Fault trees can help to recognize interrelationships between fault events.
  - SAPHIRE evaluates the fault tree to find system minimal cut sets and the system failure probability.

♦ Fault trees consist of logic gates and basic events as inputs into the logic gates.

Logic Gates

Represent the Boolean operation (e.g., union, intersection) of the input events.

Basic Events

Represent a fault such as a hardware failure, human error, or an adverse condition.

1.7. Accident Sequence Quantification Steps

1. Link fault models to the event tree sequences.
2. Evaluate each accident sequence for minimal cut sets.
3. Quantify the accident sequence minimal cut sets with event data.
4. Add operator recovery actions and common-cause failures (if not already in the fault tree and event tree logic models).
5. Determine the dominant accident sequences.
6. Partition the accident sequences into appropriate plant damage state bins.
7. Perform sensitivity, importance, and uncertainty analysis on the accident sequences.

1.8. Installation of SAPHIRE

To install:

◊ Input the media containing the SAPHIRE executable (i.e., memory stick).
◊ Double click on the SAPHIRE executable (SAPHIRE-7-27.exe).
◊ Follow the installation program instructions.

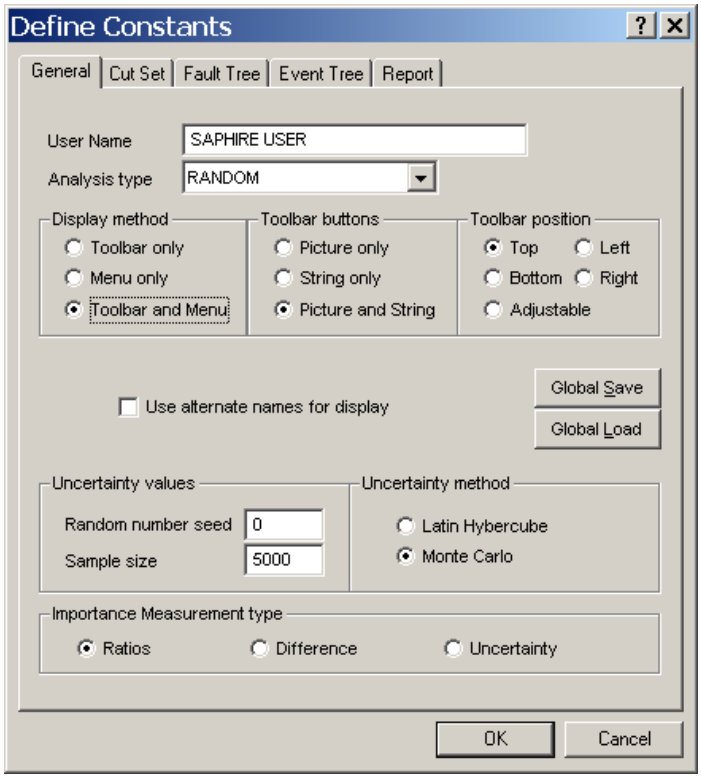
♦ The installation program will make a subdirectory on your hard drive to store SAPHIRE.

◊ Databases (such as the DEMO database) can be contained in any subdirectory that is chosen (e.g., C:\DEMO or C:\Saphire7\DEMO).
◊ The database subdirectory will contain the relational database files.
- *.IDX files contain data indices.
- *.BLK files contain variable length data (e.g., cut sets).
- *.DAT files contain actual data and data pointers.

1.9. SAPHIRE Constants

The SAPHIRE Constants dialogs are used to customize the SAPHIRE code.

- Four constant dialogs are available.
  - ◊ "General" information default values
  - ◊ "Cut Set" information default values
  - ◊ "Fault Tree" graphics default values
  - ◊ "Event Tree" graphics default values
  - ◊ "Report" output default values
- The constants can be modified by:
  - ◊ The constants are available by selecting **Utility ➔ Constants**.



- The constants can be modified by:
  1. Using the mouse or TAB key to move to a particular field.
  2. Change the constant(s) and select the next Define Constants dialog.

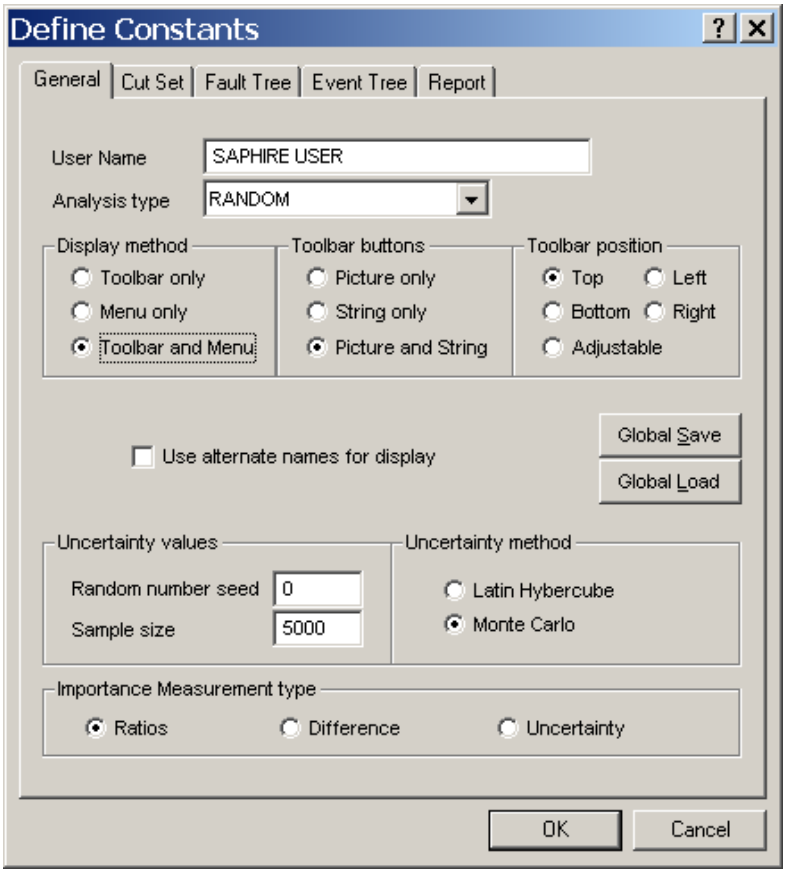
Click **OK** to save all your "constant" choices *for the project* that is currently selected. Each project may have a different set of constants.

Choose **Global Save** to save your "constant" choices to a central storage file. Choose **Global Load** to load the constants stored in the central storage file.

> ***TIP*** Once the constants are changed to the desired settings, click the **Global Save** button. Then you can retrieve the settings when you **Open** an existing project and click the **Global Load** button. For any **New** projects you create, the constants stored by the "Global Save" option will automatically be used as the initial default constants when the new project is created.

The following constant screens should be accessed and the appropriate settings made prior to creating a new Project or before clicking the **Global Save** button.

"General" constants screen

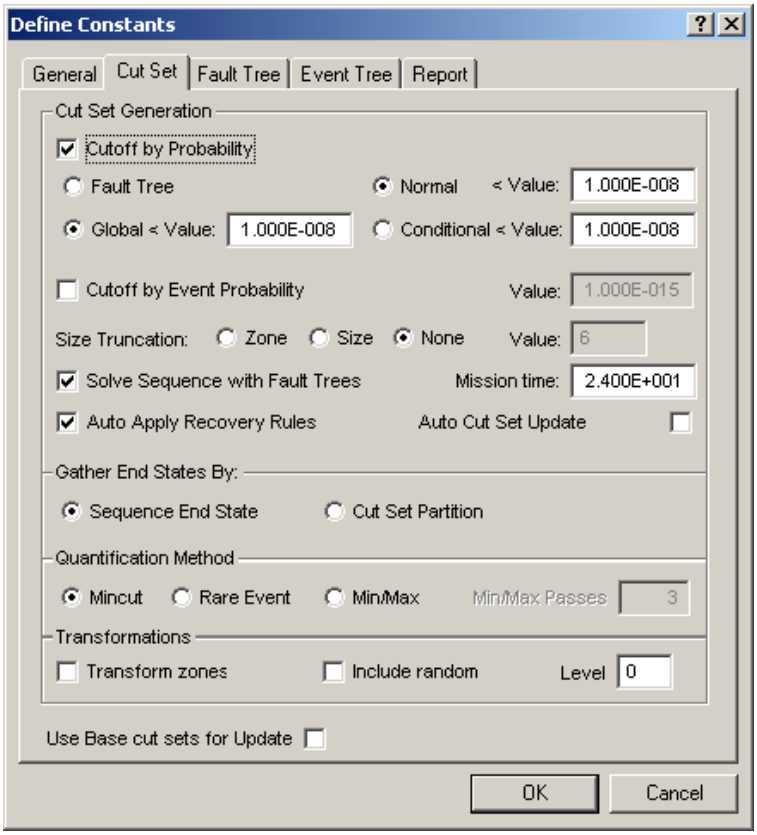


The general constant screen allows for general use information to be added.

The “Uncertainty Values” option sets the default uncertainty analysis type by selecting either Latin Hypercube or Monte Carlo radio button. Also, the random seed and number of samples can be specified.

The Importance Measurement Type defines the set of importance measures to output for the reports option.

"Cut Set” constants screen



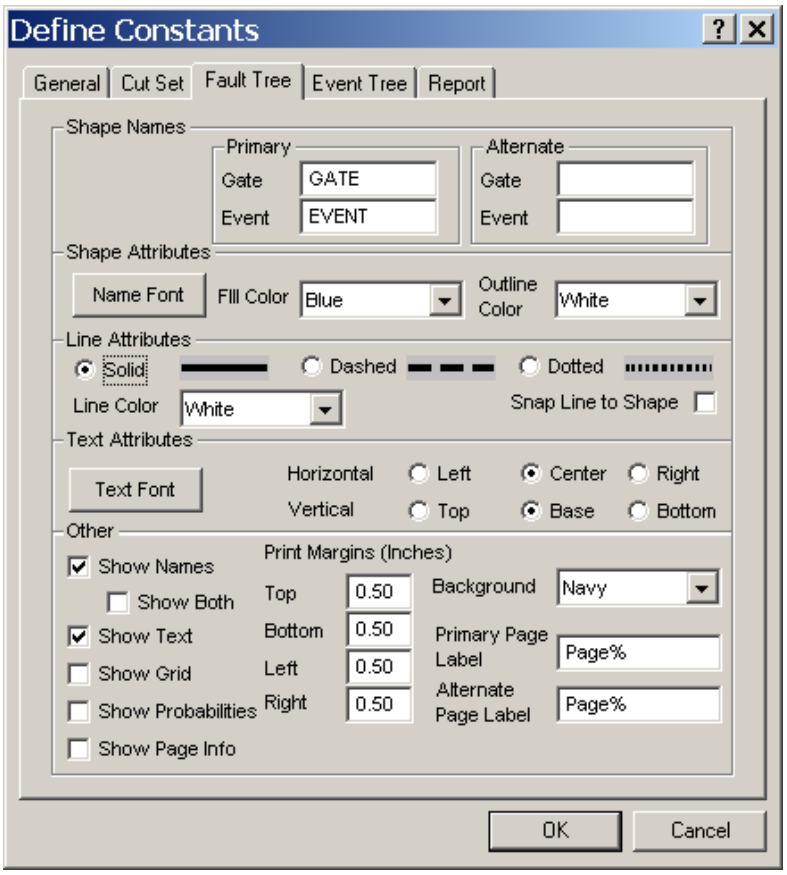
The Cut Set Generation options are used when generating both fault tree and sequence cut sets. The options marked by a check in the box and the value specified will be used when generating either fault tree or event tree cut sets.

TIP Make sure the “solve sequences with fault trees” option is checked in order to tell SAPHIRE to use the fault tree logic when generating accident sequence cut sets. Also, make sure that the “auto apply recovery rules” option is checked in order to tell SAPHIRE to automatically apply recovery rules (if they are defined) to the cut sets after generation.

The “Mission time” field specifies the default mission time. This default is used only for those basic events that have a calculation type using the mission time and that event’s mission time field is set to zero.

The “Quantification Method” option specifies which analysis method to use when quantifying the cut sets.

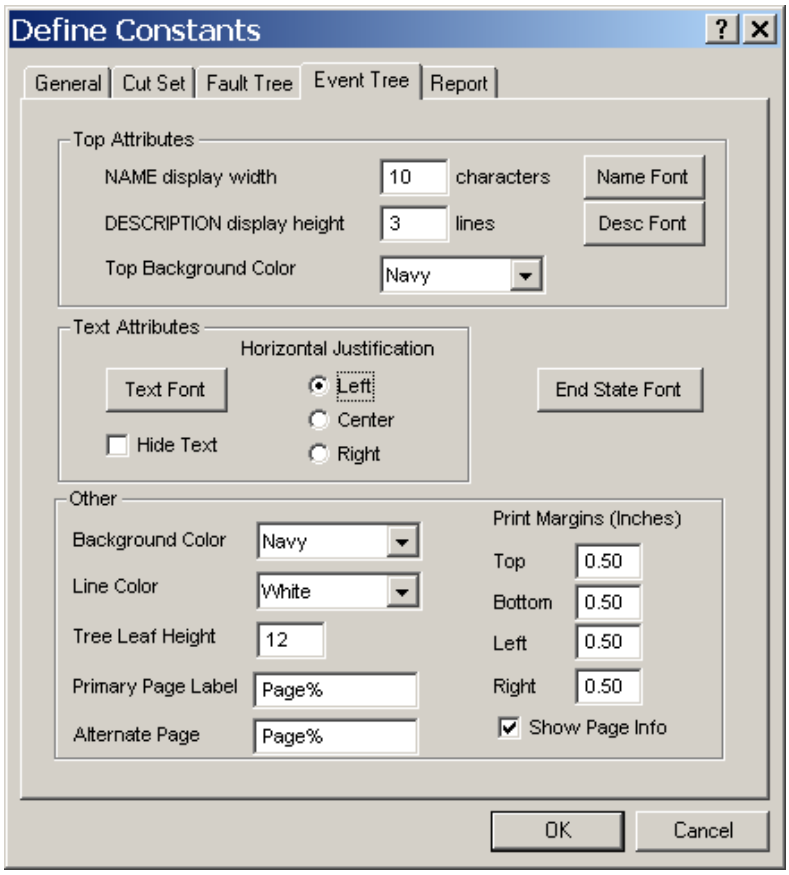
"Fault Tree" graphics constants screen



The fault tree graphic constant screen should be verified prior to creating any fault tree graphics. Check to make sure that the graphical colors that are selected will result in a readable fault tree (e.g., do not select black lines on a black background).

The constants shown above are for illustration purposes and you can change the color and font schemes to suit your tastes. The “Snap Line to Shape” option is used to simplify the graphical drawing functions of SAPHIRE. If checked, lines drawn from any part of one shape to any part of another shape will be adjusted to connect the two shapes. If this option is not checked, lines will be placed exactly where dropped.

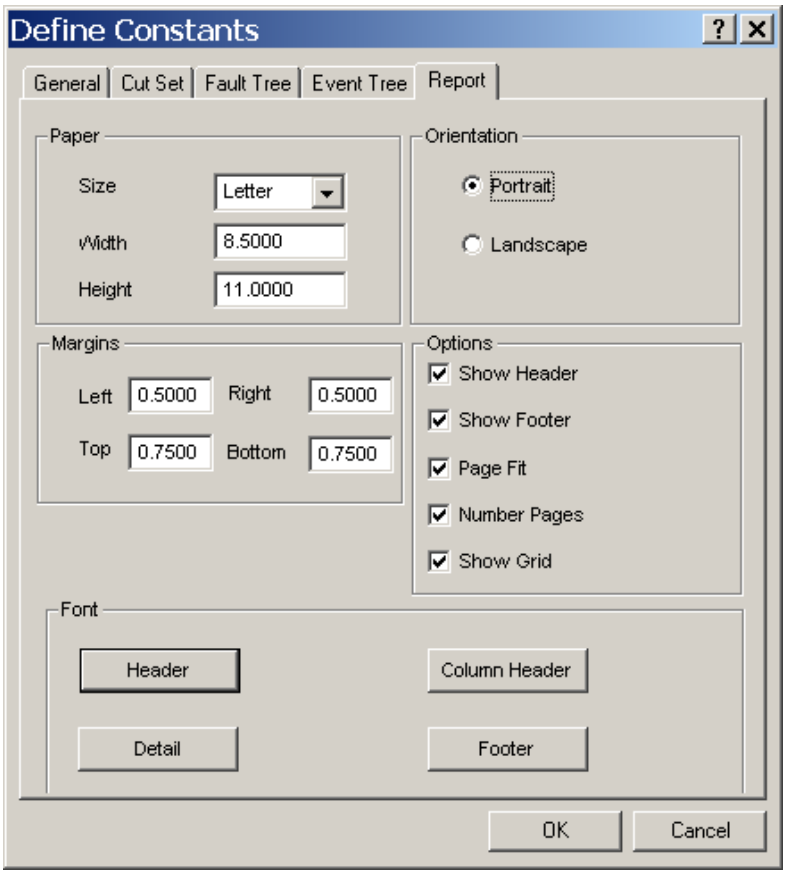
"Event Tree" graphics constants screen



The event tree graphic constant screen should be verified prior to creating any event tree graphics. Check to make sure that the graphical colors that are selected will result in a readable fault tree (e.g., do not select black lines on a black background).

The constants shown above are for illustration purposes and you can change the color and font schemes to suit your tastes.

“Report" output constants screen



The report output constant screen should be verified prior to creating any reports.

The constants shown above are for illustration purposes and you can change any of the parameters to meet your reporting needs.

1.10. Lists and Masks

♦ List Boxes –

Many dialogs in SAPHIRE contain list boxes. In some list boxes, multiple items can be selected for processing. An item in a list is selected if it is highlighted. There are various ways to select items from a list.

◊ To select a single item, click with the left mouse button on the desired item and let go of the mouse button.

◊ To select multiple continuous list items, you can click with the left mouse button the on first desired item and drag up or down the list to the last desired item and then let go of the mouse button. Alternately, click the first desired item then, holding down the ***Shift*** key, click the last desired item.

◊ To select multiple non-continuous items in the list, click several items while holding down the ***Control*** key.

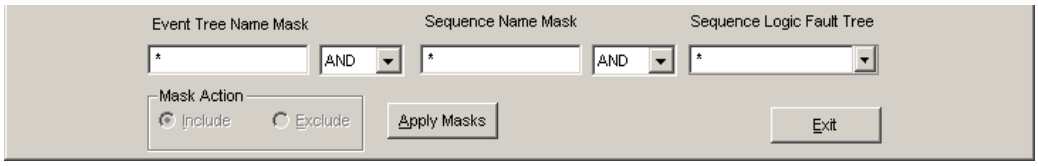
♦ Masks –

Some dialogs with list boxes provide a "Mask" capability which allows the user to select items from the list based on matched criteria. Generally, the mask is applied to the name of the item (e.g. Fault Tree name or Event Tree name). To use the mask capability,

1. In the mask entry field type the common characters of the names you wish to match. The wildcard characters, asterisk (*) and question mark (?), can be used in the mask. The asterisk represents one or more characters that a group has in common. The question mark represents a single character in that position of the string that a group has in common.

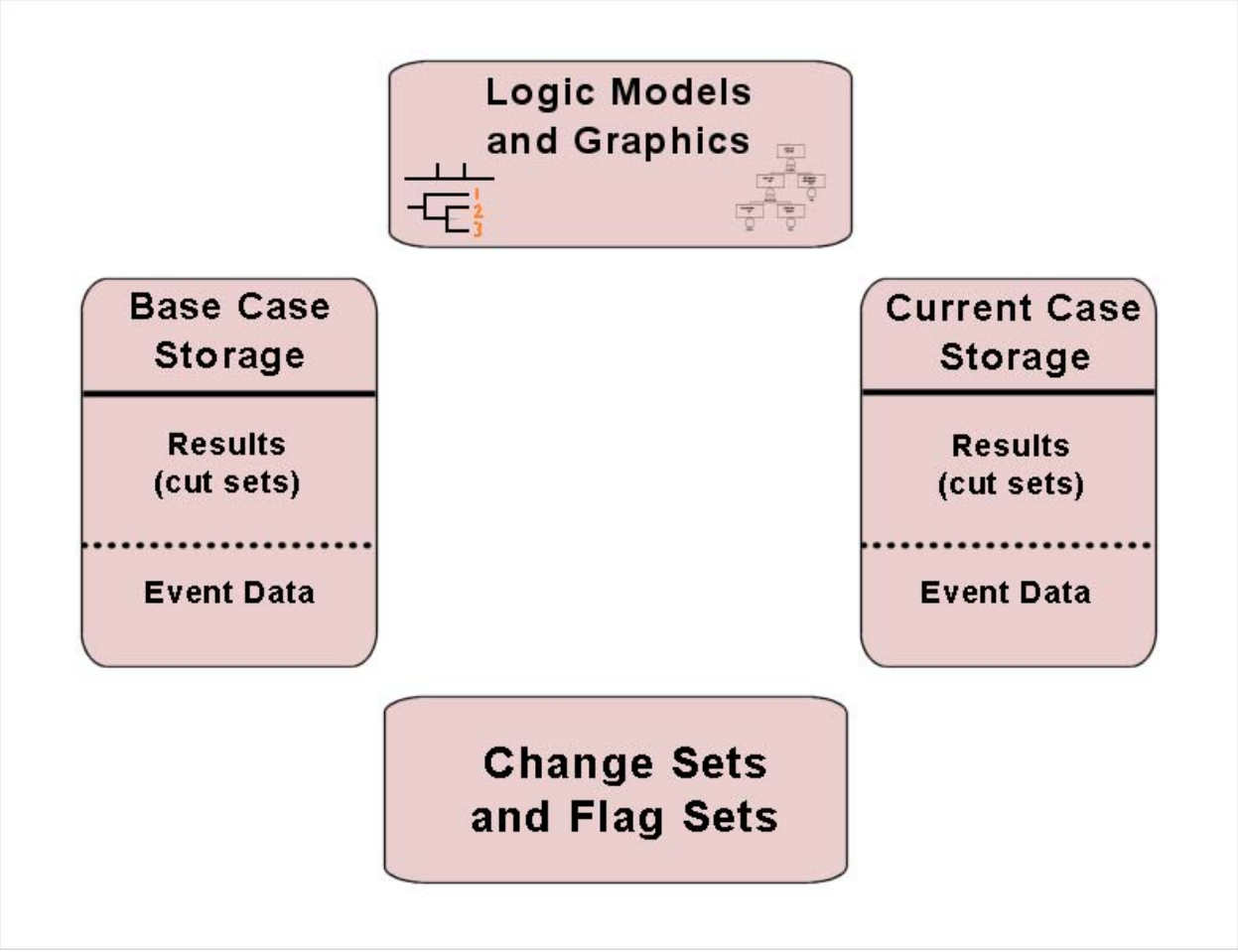
2. Click either the Include or Exclude radio button, depending on whether you want these items included in the selection or excluded from it.

3. Choose the Apply Mask button. All list items with names matching your mask will be selected/deselected.



2 DATABASE CONCEPTS

Section 2 presents an overview of the SAPHIRE database structure. Included in this section are discussions of SAPHIRE projects, base case versus current case, base case updates, and change sets.

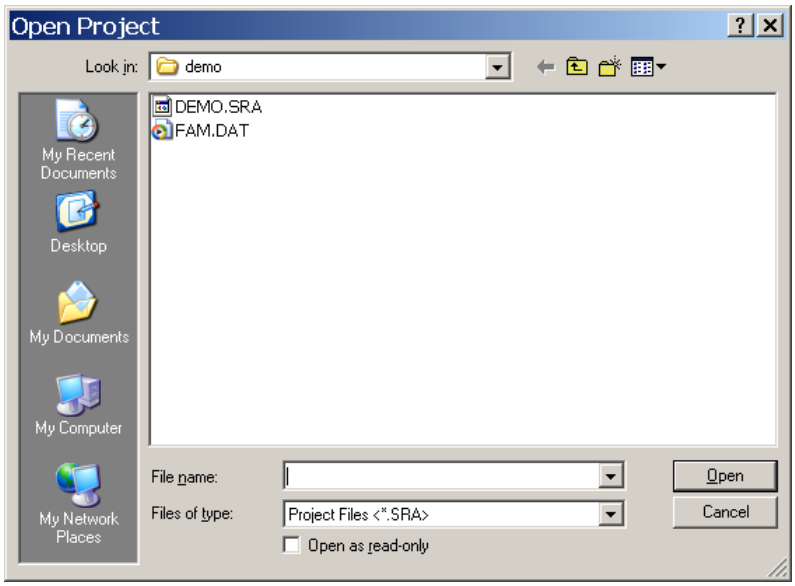


2.1. SAPHIRE Projects

♦ In SAPHIRE, the term “project” represents a single, specific database.

To select an existing project:

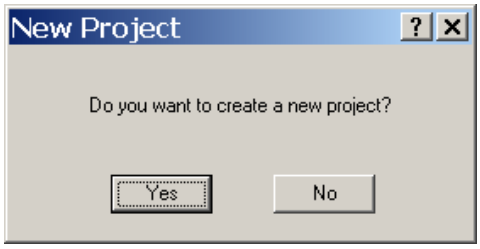
1. Use the **Open** icon or use the **File → Open Project** option.
2. An “Open Project” window will appear. Use the various Window Explorer options to find the folder containing the existing project.



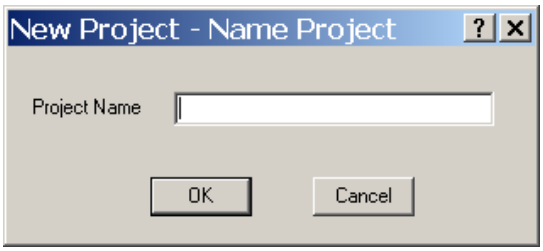
3. Select and open either the *.SRA or FAM.DAT file in the opened folder.

To create a new project:

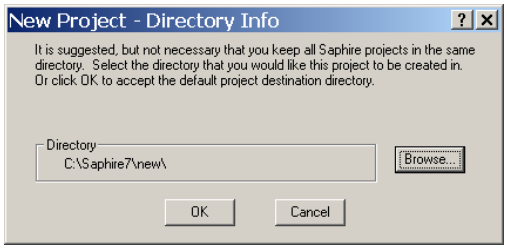
1. Select the **New** icon or use the **File → New Project** option.
2. A “New Project” window will appear. Click **Yes**



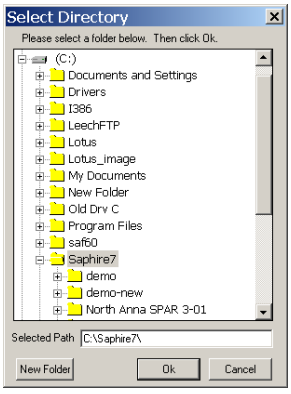
3. A “New Project – Name Project” window will open. Type the new project name and click **OK**.



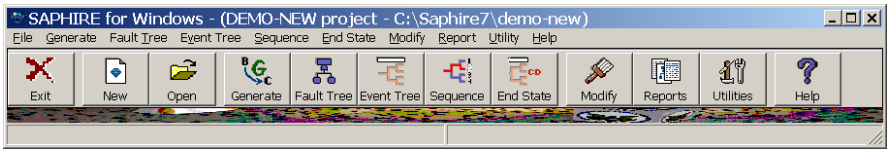
4. A “New Project Directory – Info” window will open. If the location is as desired, Click **OK**. If a different location is desired, click **Browse**.



5. A “Select Directory” window will open. Use the **New Folder** button or browse to find the desired folder for the new project. Click **OK**.



6. Final step will be to confirm the desired folder entered. Click **OK**. The main SAPHIRE menu will be displayed and the project name and the project location will be shown at the top of the main SAPHIRE menu.

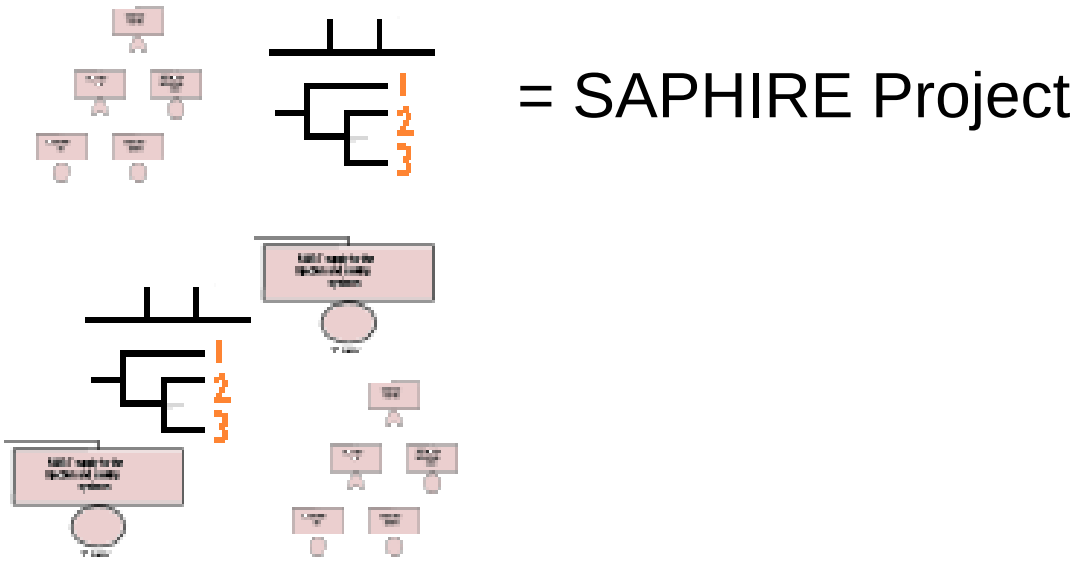


- Modifications to a database (e.g., a new fault tree is developed) are always made to the currently selected project.

- For a given project, only one list is kept for all types of information. Thus, within a project, only a single copy of a particular fault tree, event tree, or basic event is ever stored in the database.

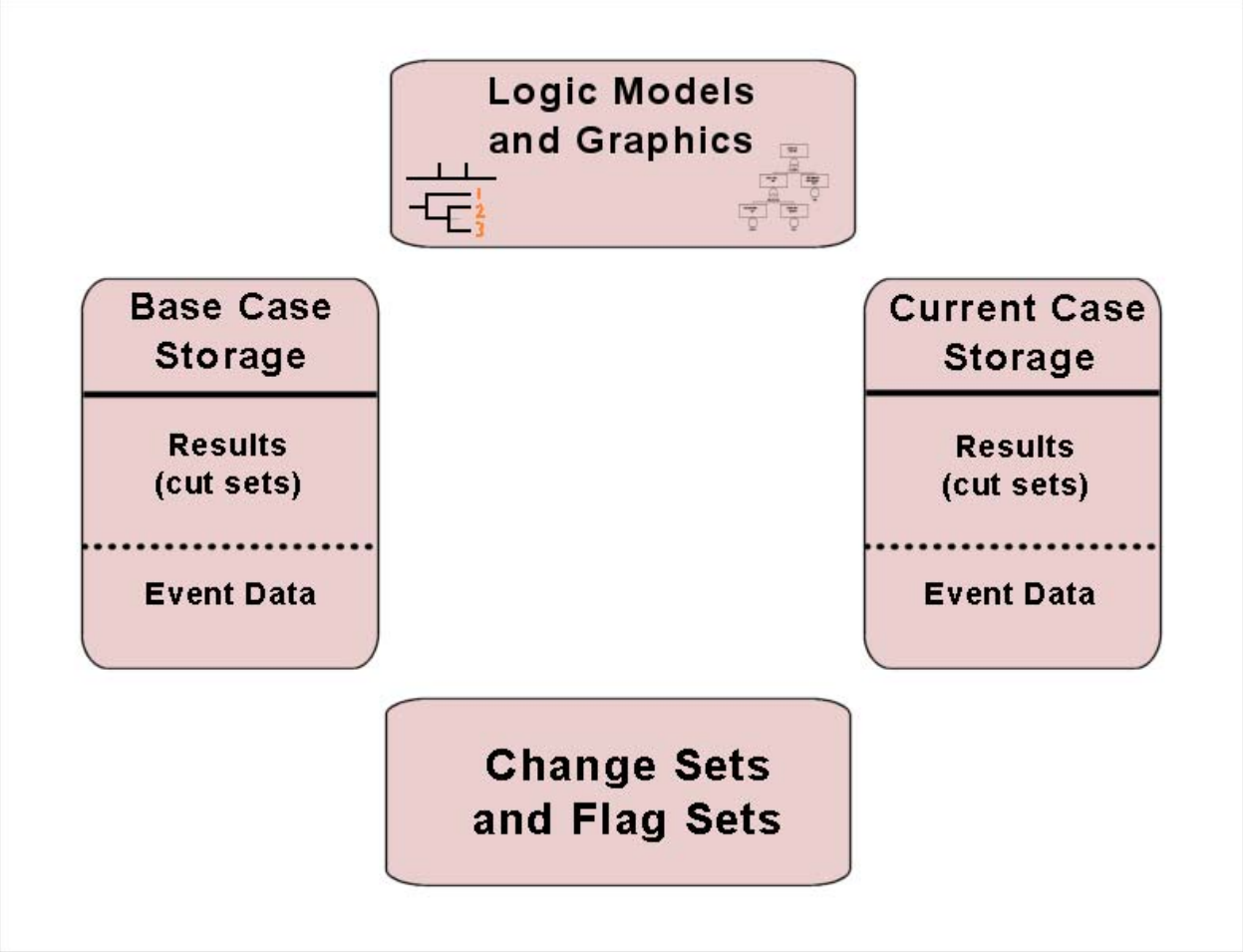
Project (Definition)

A group of fault tree logic and graphics; event trees and sequences; basic events and related data; cut sets; analysis results; and descriptions.



2.2. Base Case Versus Current Case Data

♦ Base case and current case are two separate parts of a project database.



◊ **Base Case** data is stored in the data base files as a “permanent” record
◊ **Current Case** data is used to perform an analysis (e.g., cut set generation and quantification)

The Current Case Is

◊ Created (via the **Generate** option) by applying change sets to base case data
◊ Used for sensitivity or event analysis

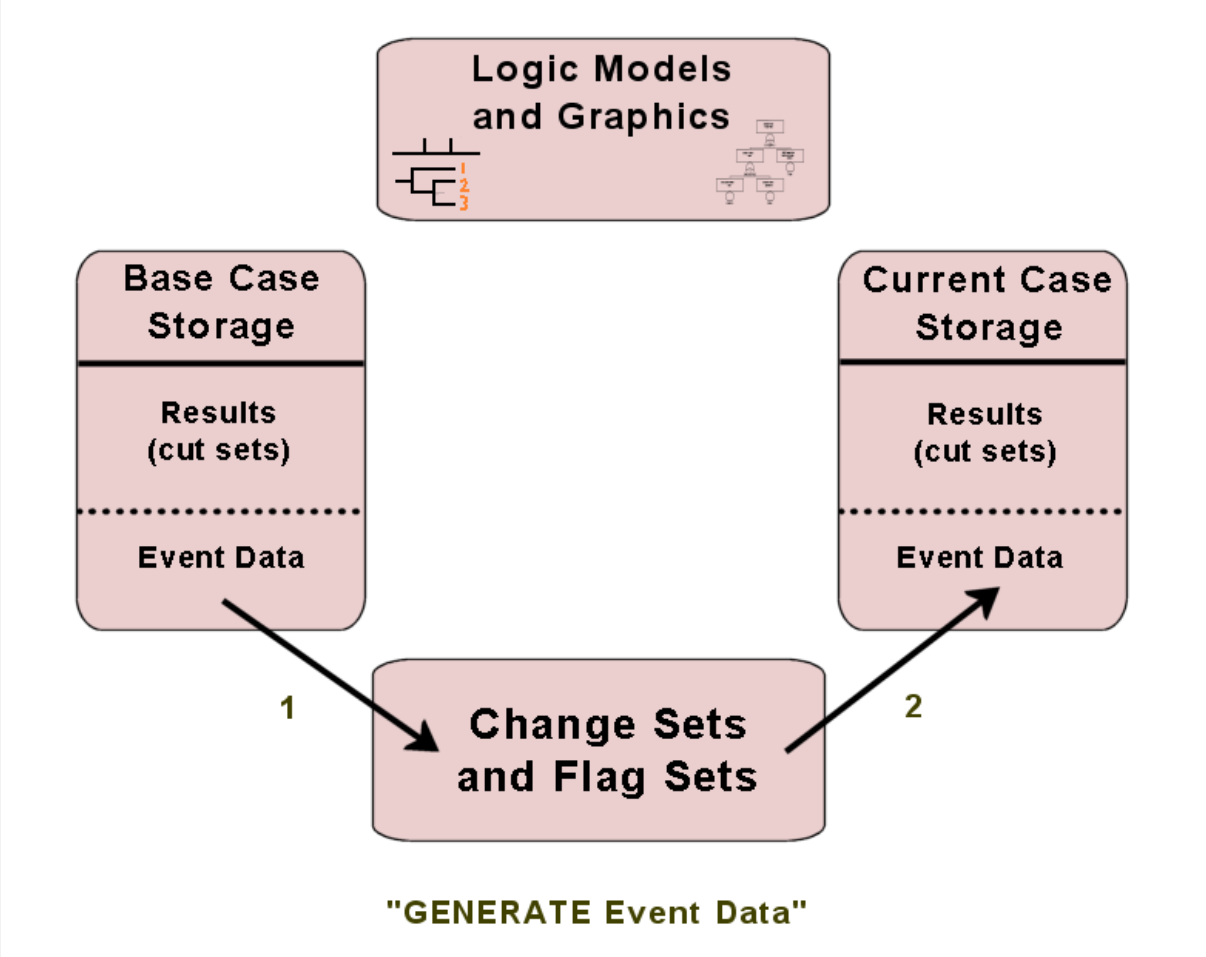
♦ All SAPHIRE calculations use the data stored in the **current** case.

♦ Current case can equal the base case in order to reproduce the original study stored in the base case.

2.3. Generating Event Data

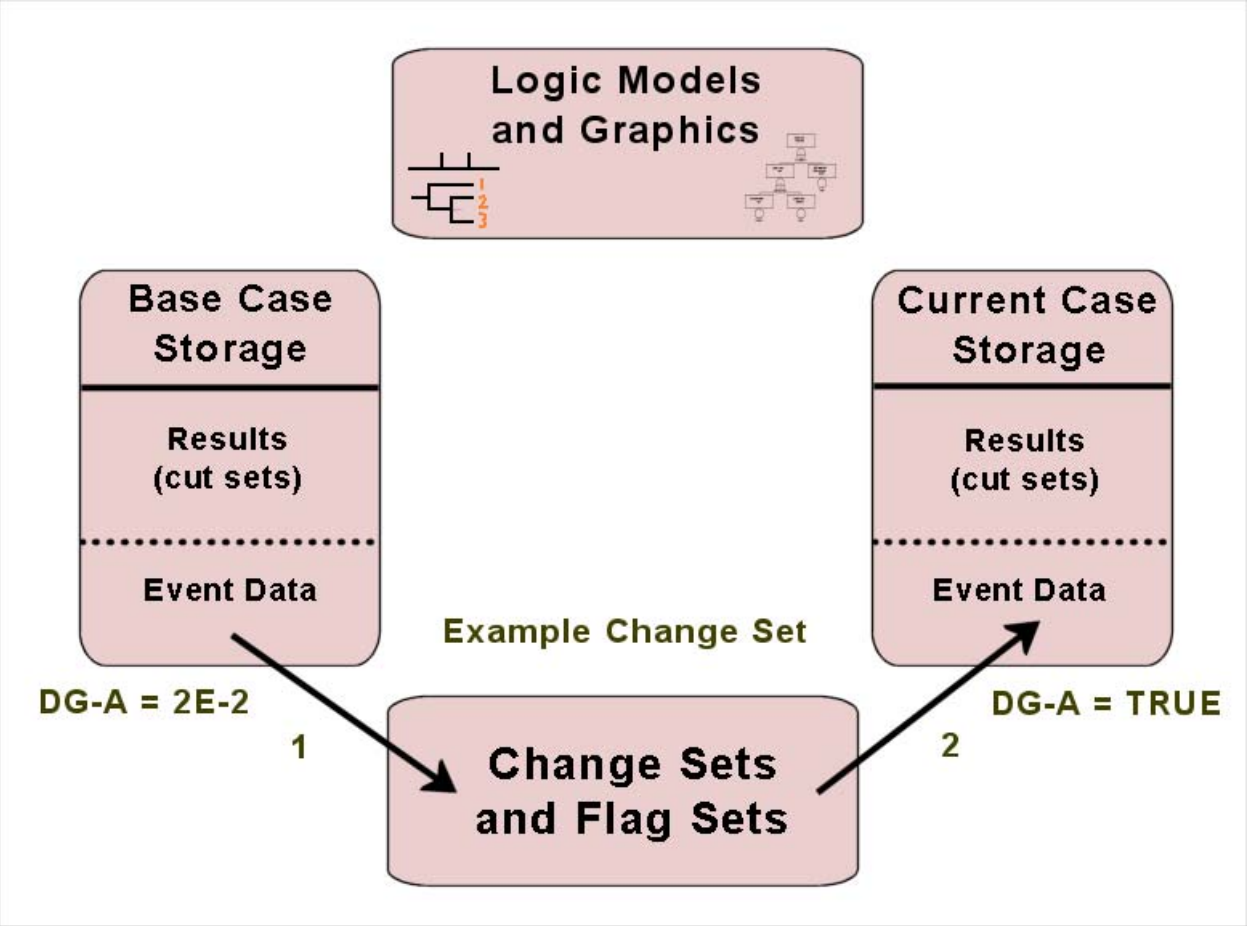
The **Generate** option transfers base case data to the current case (after making changes specified in any marked change sets). SAPHIRE *always* uses current case data for analysis.

If the base case data is changed and the **Generate** option is *not* performed, the data that is used for the analysis may not reflect the changes. After changing basic event data (in the Base Case, Change Set, or Flag Set), performing the **Generate** option ensures the current case data reflects these changes.



Change Sets (Definition)

Change Sets are a user-defined set of changes (think data filter) that will be applied (on the base case data) when data is transferred to the current case via the **Generate** option. Multiple change sets can be defined and applied singly or in combination.



2.4. Rules for Creating and Using Change Sets

- No limit to the number of change sets that can be added to the data base.
- Change set name is limited to 24 characters, the description is limited to 120 characters.
- A change set can contain one class change and unlimited individual probability changes.
- Multiple change sets can be used in combination to create different sensitivity studies.

EXAMPLE: Two change sets are developed. The first is named "A" and sets all valves to failed. The second is named "B" and sets all pumps to failed. The possible scenarios are

Change set(s) that are marked	Sensitivity case
None	Original base case data
A	Valves failed
B	Pumps failed
A and B	Both valves and pumps failed

The order of "marking" a change set is important. (Change sets are marked by double-clicking the line containing the change set.)

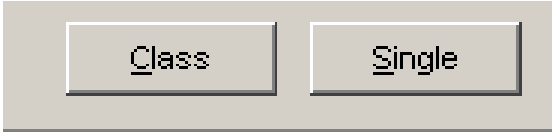
- ◊ The first selected change set will be the first one that is applied.
- ◊ Later changes will overwrite earlier ones if there is any overlap.
- ◊ A particular change set may include both a Class change and Single changes. The Class change is applied first and then the Single changes are applied second. Thus, the individual probability changes will overwrite a class change if both types are in a particular change set.

♦ Base case data and changes made to the current case can be viewed by using the **Generate → Report** option.
- ◊ Unaffected events *(those events not modified by a change set)*
- ◊ Affected events *(those events which are modified by a change set)*
- ◊ All events *(all events)*

Class Changes

Class changes use a basic event attribute to search for a class of basic events to which the defined change applies

- The search criteria are defined first
- The change to be applied is then defined



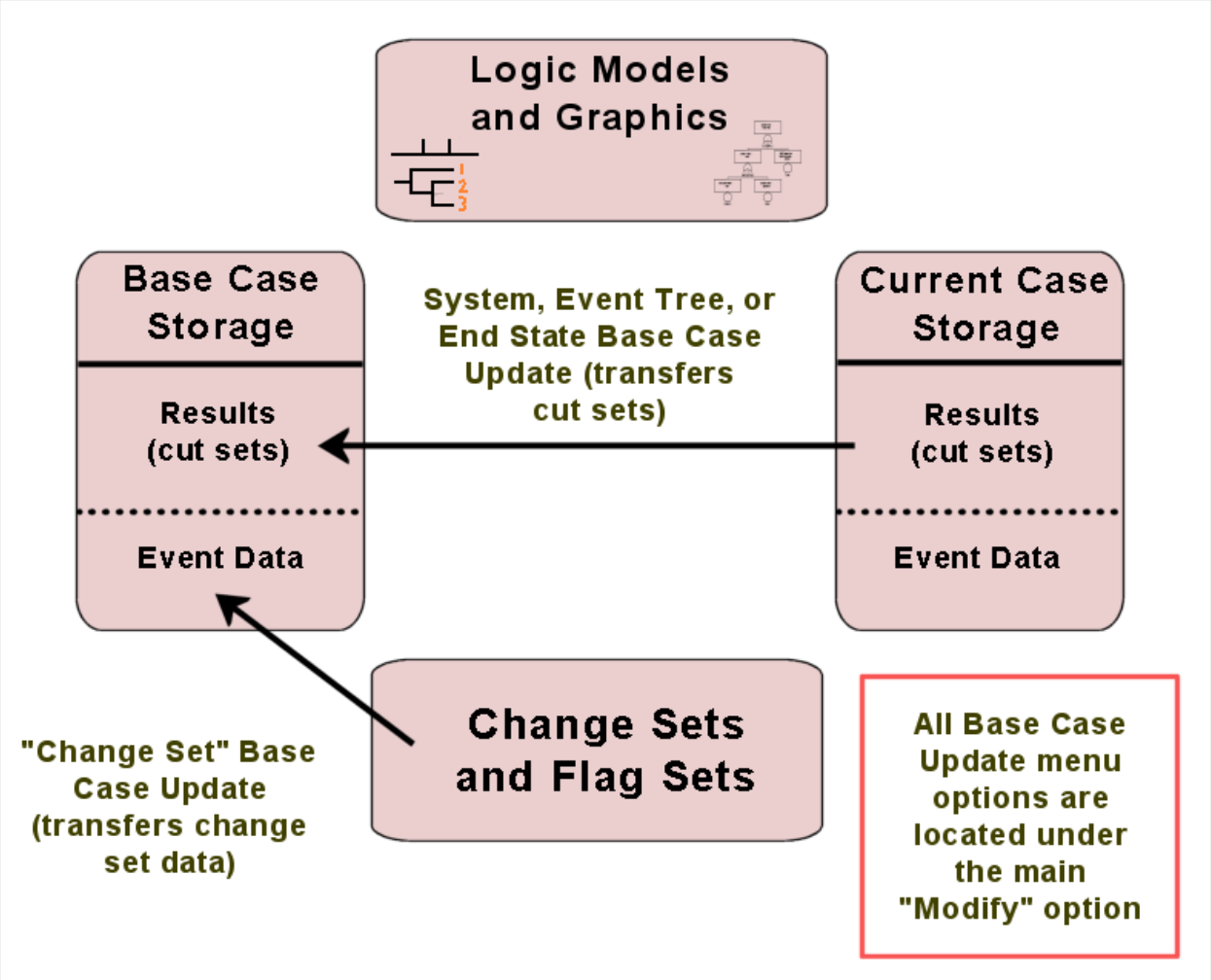
Single Changes

Single changes only modify individual, user-identified basic events

- The desired basic event is selected
- The changes to the basic event are then defined

2.5. Base Case Update

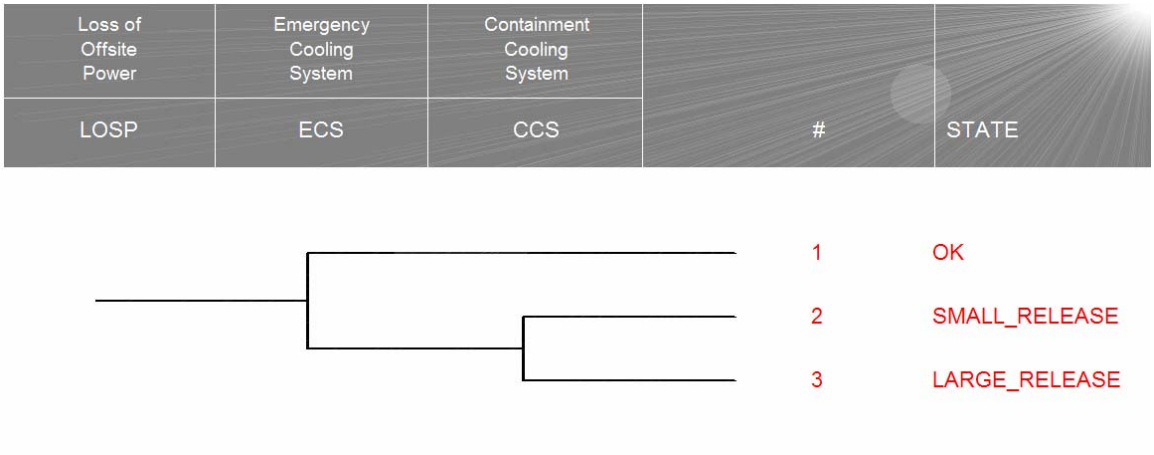
- Base case data and results are changed by **updating** the base case. Updating the base case transfers the current case data or results into the base case.
  - **Base Case** results are stored in the data base files as a “permanent” record



Notes

3 BUILDING EVENT TREES

Section 3 introduces event tree terminology and SAPHIRE event tree modeling conventions. You will learn how to enter an event tree into SAPHIRE using the graphical event tree editor and also how to edit an existing event tree.



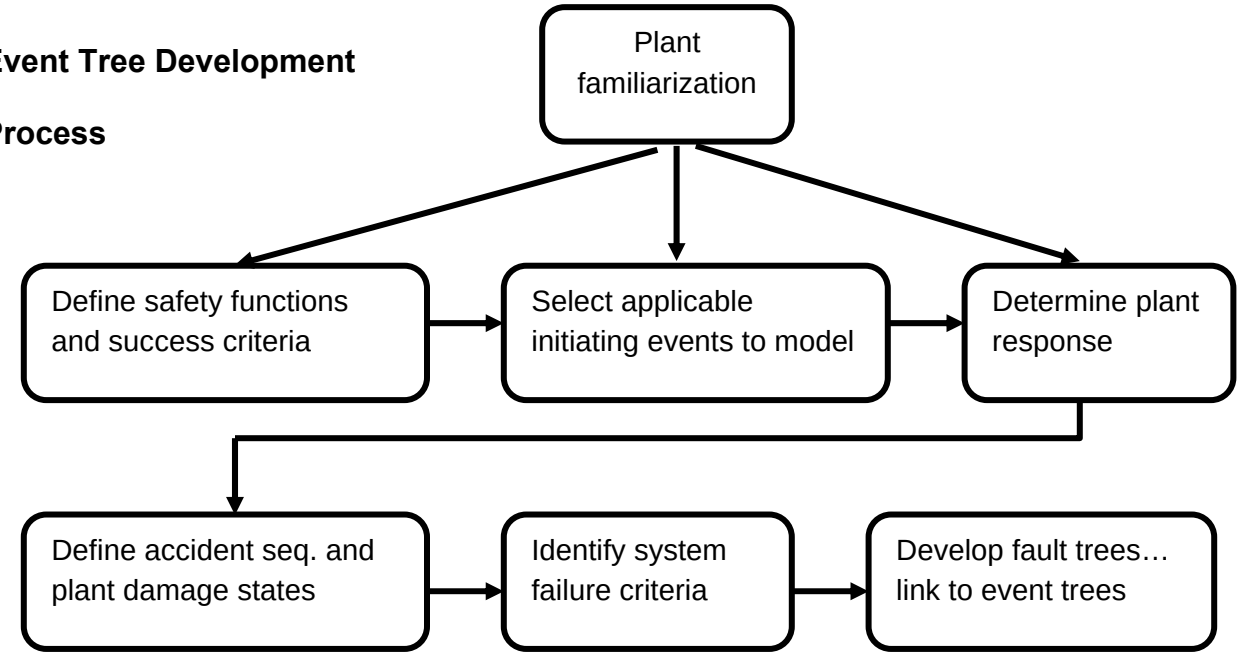
3.1. Event Tree Development

Event trees are developed by starting with an initiating event and branching to the right as various safety functions are questioned for success (up branch) or failure (down branch).

Event trees provide a traceable way to perform the following functions:

- ◊ Identify accident sequences
- ◊ Identify essential safety system functions
- ◊ Quantify sequence frequencies

Event Tree Development Process



Event Tree Definition

Event trees are logical representations of significant plant responses to initiating events with each sequence resulting in either a safe condition (such as safe shutdown) or an accident condition (such as core damage).

3.2. Event Tree Terminology

Initiating Event
An initiating event is an operational occurrence (such as a LOCA or transient) which threatens fuel safety and may require safety system response to avoid core damage.

Top Event
Safety systems (or human actions) which are intended to respond to the initiating event.

Branching
The branching underneath a top event which indicates success with an up branch and failure with a down branch.

Pass
When there is no branching beneath a top event, then the top event is not relevant to the particular sequence.

Sequence
The branching path, from initiating event to end state, which is a unique combination of system failures and/or successes.

End State
Groups of accident sequences which share certain characteristics that the analyst delineates. These may be related to ability to perform a safety function or timing of functional failures.

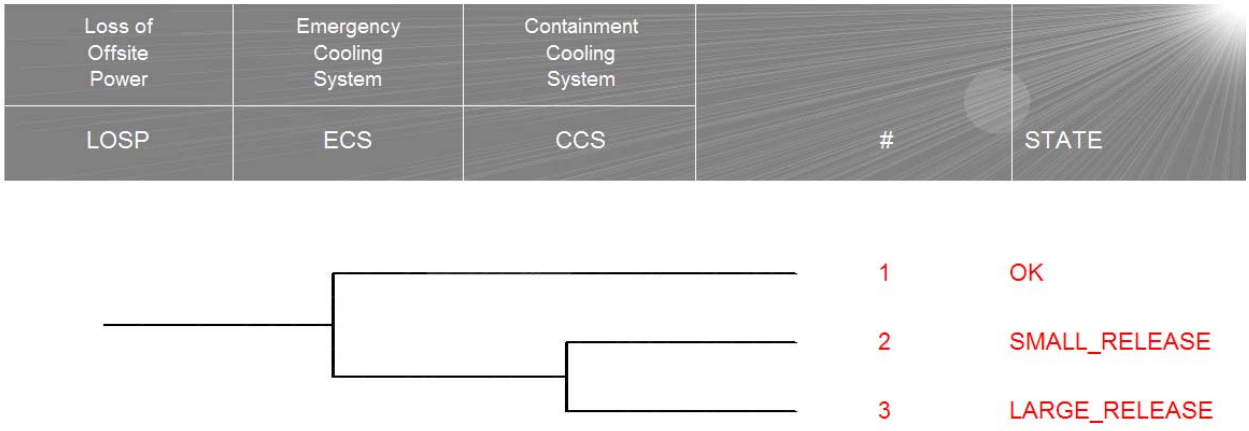
3.3. SAPHIRE Event Tree Conventions

Event Tree Names	—	Event tree names may be up to 24 characters. The event tree name **does not** have to be the same as the initiating event name.
Top Event Names	—	24 characters allowed.
Initiating Event	—	Only one initiating event allowed per event tree.
OK, @, Success	—	If the end state column entry is **OK**, begins with **@**, or **Success**, then logic for that sequence will not be developed, and the sequence will not be analyzed.

Branch	—	SAPHIRE always uses success for the up branch and failure for the down branch.
Transfer Trees	—	An event tree can branch to another event tree by using transfers. You must indicate that a transfer is to be invoked and specify the transfer event tree file name. The first top event in the transfer tree is ignored by the calling tree.

NOTE: Do not use ***, ?, \, @, /*, **or space** in naming SAPHIRE event trees or top events.

Example Event Tree



3.4. Event Tree Graphics Features Guide

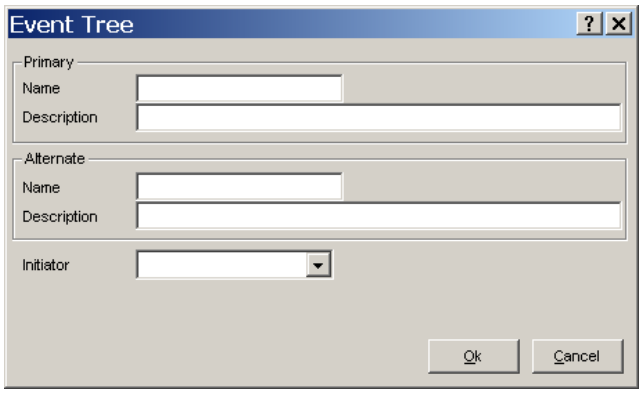
This section provides an overview of a variety of activities one may wish to perform when creating and editing event trees. Included in this section is information on the following:

Beginning the event tree editing session

To add a new event tree

Select Event Tree from the menu bar or click the **Event Tree** icon.

Event Tree List window will open. Select the **Add** icon or right mouse click and select **Add Event Tree**.



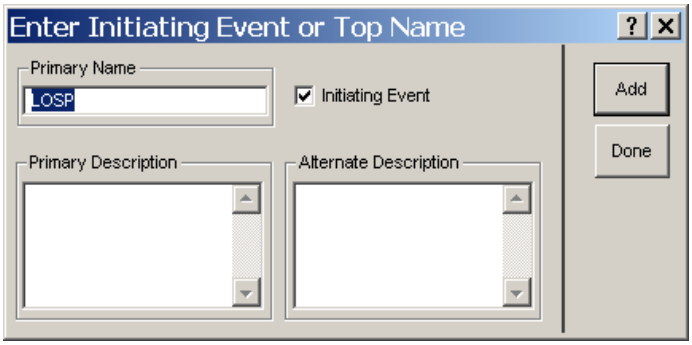
Enter the event tree name, description, and initiator. Press **Ok**.

The event tree will be added to the *Event Tree List* window. Now highlight the desired event tree and click the **Graphics** icon or left mouse click and select **Edit Graphics**. Another option would to be double to left mouse click over the desired event tree.

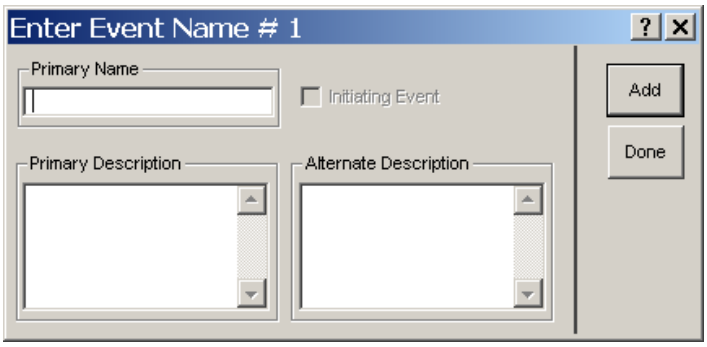
Note – Another method to add and begin an editing session is to open the *Event Tree List* window. Then select the **Graphics** icon to or right click and select **Edit Graphics**. Next, from the menu bar select **File ➔ New**.

Now edit the event tree…

Event Tree List dialog pop-up menu: **Edit Graphics**



Enter the initiating event name and check the Initiating Event box. (If creating a transfer tree, you may deselect the check box when the first event is not an initiating event.)



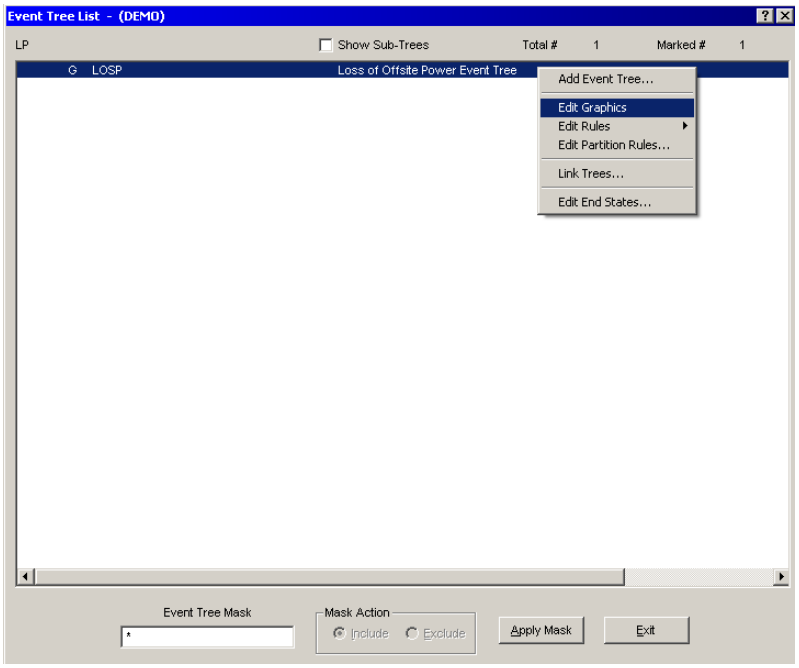
Continue by specifying top event names (from left to right on your event tree). When you have entered all the top events, choose "Done."

To edit an existing event tree

SAPHIRE menu bar: **Event Tree**

Event Tree List dialog: Highlight the desired event tree name.

Event Tree List dialog pop-up menu: **Edit Graphics**



Editing event tree branches

To add branches to event trees

Select the branch point then add a branch by either

Event Tree Editor menu bar: **Edit ➔ Add Branch Above**,

OR Choose the **Add Branch Above** button on the tool bar.

A new branch will appear in the tree above the selected branch.

To delete an unwanted branch

Use the mouse to select the unwanted branch

Event Tree Editor menu bar: **Edit ➔ Delete**

Editing top events

To add a top event, choose the Top object button from the button bar. The cursor will change to the Top cursor +TOP

Move the cursor to the top in the header that will follow the top you are adding. To add a top after the rightmost top, move the cursor to the right of the last top. Click the mouse. The *Edit Event* dialog will be displayed

To delete a top event

Select the top using the Pick cursor.

Choose Delete from the Edit menu or press the Delete key

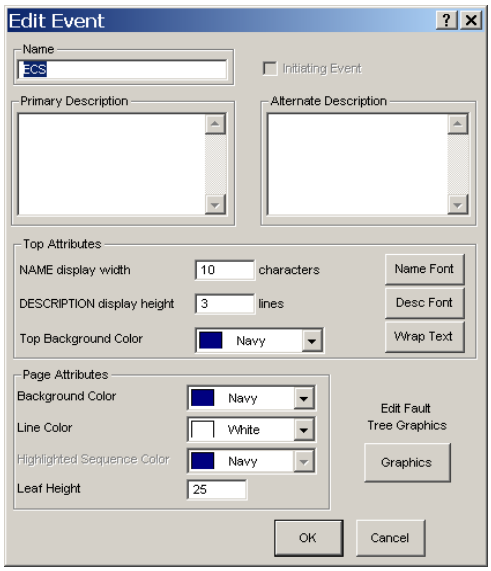
To modify top event

Select the top using the Pick cursor.

Right-click. The *Edit Event* dialog will be displayed.

The Edit Event Dialog

This dialog is displayed any time a top is added or modified.



Name - The name of the top. Maximum of 24 alphanumeric characters.

Initiating Event - Select this check box if the top is an initiating event. (Only enabled on the first or leftmost top.)

Primary Description - Brief description of the top. Maximum of 120 alphanumeric upper and lowercase characters.

Alternate Description – Alternate brief description of the top. Maximum of 120 alphanumeric upper and lowercase characters.

Top Attributes

NAME display width - The number of characters of every top name to display.

DESCRIPTION display height - The number of lines of every description to display.

Name Font - The font size of every top event name.

Desc Font - The font size of every top event description in points (1/72 in.).

Wrap Text – Automatically wrap description text if text is wider than the name display width.

Top Background Color – The background color for the top header.

Page Attributes

Background Color - The background color for the event tree screen.

Line Color – The branch line color.

Leaf Height – The vertical spacing of branches, approximately in points.

Graphics – Open the fault tree graphic editor for the selected top.

OK - Apply the modifications to the top and close the *Edit Event* dialog.

Editing or adding end states

To modify sequence columns (or turn on end state and sequence headers)

Choose the Pick button.

Position the Pick cursor over the sequence header field.

Right-click. The *Edit Sequence Header* dialog will be displayed (see figure below).

To enter either end state names or sequence names

Select the desired sequence branch using the Pick cursor left-click.

Right-click. The *Edit Sequence* dialog will be displayed (see next page)

To move a header on the graphic (for example, to shift the end state column to the left)

Choose the Pick button.

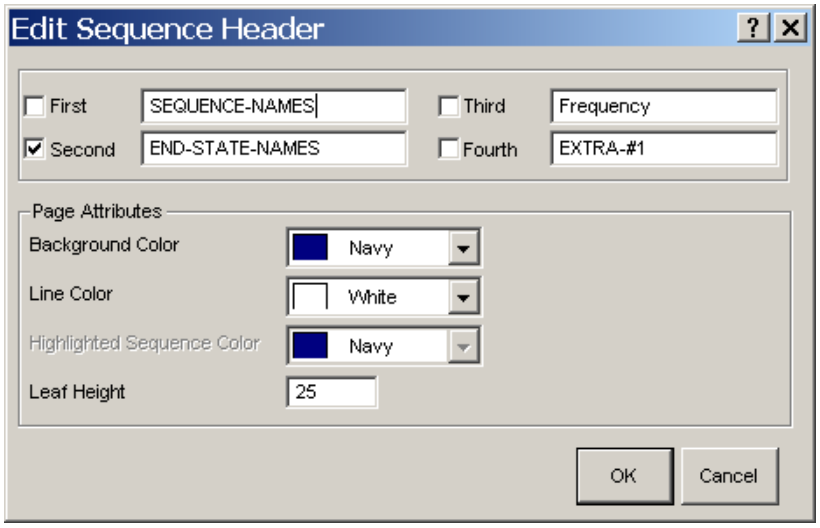
Position the Pick cursor over line to the left of the desired column.

The cursor will change to the Header cursor.

Drag the mouse to the desired location. The cursor will change to the Move Column cursor. Release the mouse and the column will be moved.

Edit Sequence Header Dialog

Change the sequence information column header in the diagram.



First - The sequence name. Maximum 24 alphanumeric characters. This column stores a sequence "name" and is optional. Select the check box to display this column.

Second - The sequence end state. Maximum 24 alphanumeric characters. This column will contain the end state or transfer tree names and is required. Select the check box to the left of this field to display this column.

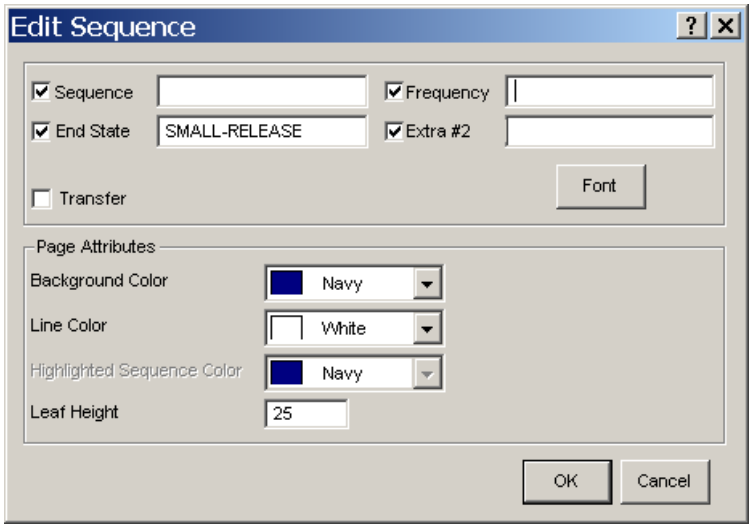
Third -The header for the third sequence column. Maximum 24 alphanumeric characters. This can be used for whatever information you wish, however, options exist elsewhere in the program that will put the sequence frequency of this sequence in this column. Select the check box to the left of this field to display this column.

Fourth -Extra information for the sequence. Maximum 24 alphanumeric characters. This can be used for whatever information you wish. Select the check box to the left of this field to display this column.

OK - Close the *Edit Sequence Header* dialog and apply the changes the diagram.

Edit Sequence Dialog

Each sequence path has some additional information that can help define it. This information includes the name of the sequence, the end state of this sequence, its frequency, and a user defined field. If the sequence continues through another event tree then the **end state field** is the name of the transfer tree. You have an option to display this additional information and change its location.



Sequence - The name of the sequence (optional). Maximum of 24 alphanumeric characters.

End State - The name of the sequence end state. Maximum of 24 alphanumeric characters.

Frequency - Extra information column #3 for the sequence (optional). Can contain the frequency for the sequence. Maximum 24 alphanumeric characters.

Extra #2 - Extra information column #4 for the sequence (optional). Maximum 24 alphanumeric characters.

Transfer - Check this box if the sequence continues in another event tree. The end state field contains the name of the transfer event tree.

Font - The font of the information text.

OK - Close the *Edit Sequence* dialog and apply the changes the diagram.

Adjust the visual display of the event tree

To change the view of the event tree

From the *Event Tree Editor* menu bar select **View ➔ Zoom**.

OR

Select the **Zoom** button from the button bar.

OR

Press the **Ctrl+Z** key combination.

The cursor is changed to the Zoom cursor . 

Move the cursor to the desired location.

To **zoom in** or magnify the diagram by 40%:

Click the mouse. The diagram will magnified by 40%.

To **zoom out** or reduce the diagram by 40%:

Right-click the mouse. The diagram will be reduced by 40%.

Use the scroll bars to change the portion of the diagram displayed in the window.

To redraw the diagram

Event Tree Editor menu bar select **View ➔ Refresh** or click the "fit to page" button to re-center the fault tree.

Place descriptive text on a specific event tree branch

To enter text on a specific branch (text will stay with the branch when event tree is "fit to page." Select the desired branch, right click and select edit text. Enter the desired text and click **OK**.

Place descriptive text on the event tree graphic

To enter text (for a title or other description)

Choose the Text object from the tool bar. The cursor will change to the Text cursor . 

Position the text cursor at the desired location.

Click the mouse. The *Edit Text* dialog will be displayed.

To move text

Select the desired text using the Text Pick cursor . 

Drag the text to the new location.

To modify existing text and attributes

Select the desired text using the Text Pick cursor.

Right-click. The *Edit Text* dialog will be displayed.

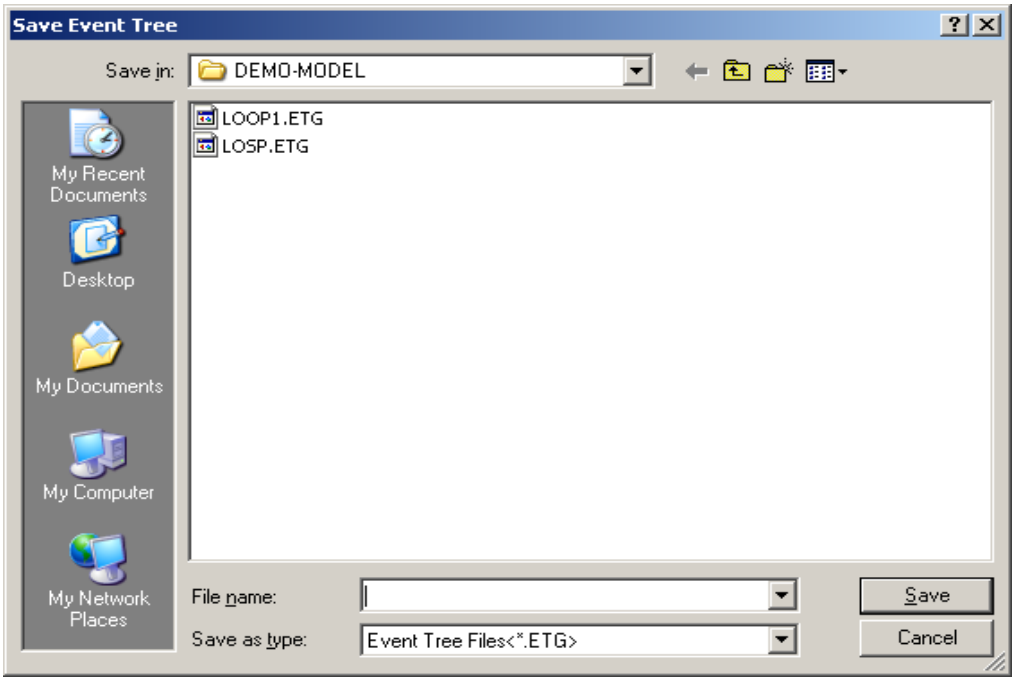
Ending the event tree editing session

To save the event tree

Event Tree Editor menu bar: **File ➔ Save**,

OR Choose the **Save Diagram** button on the tool bar

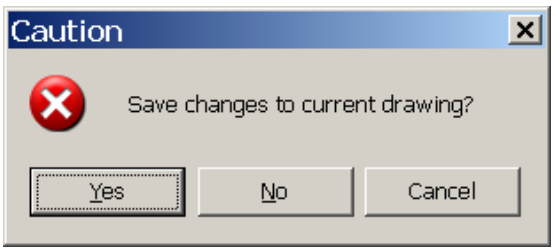
Name the event tree file as directed.



To exit without saving

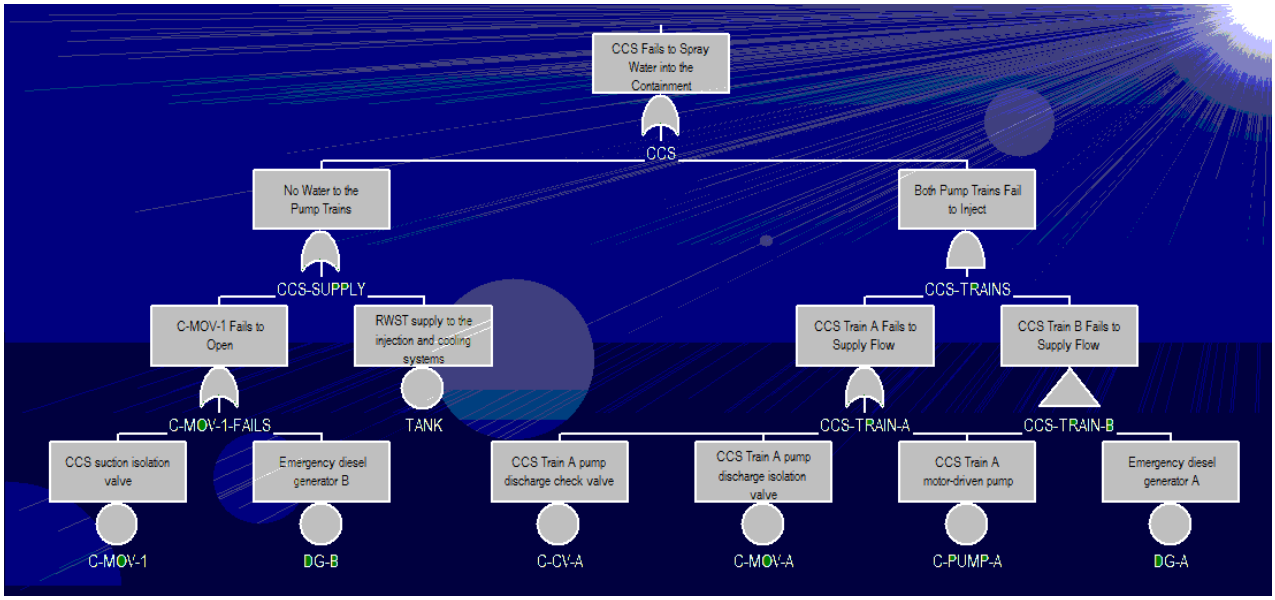
Event Tree Editor menu bar: **File ➔ Exit**

Choose **No** to quit without saving.



4 BUILDING FAULT TREES

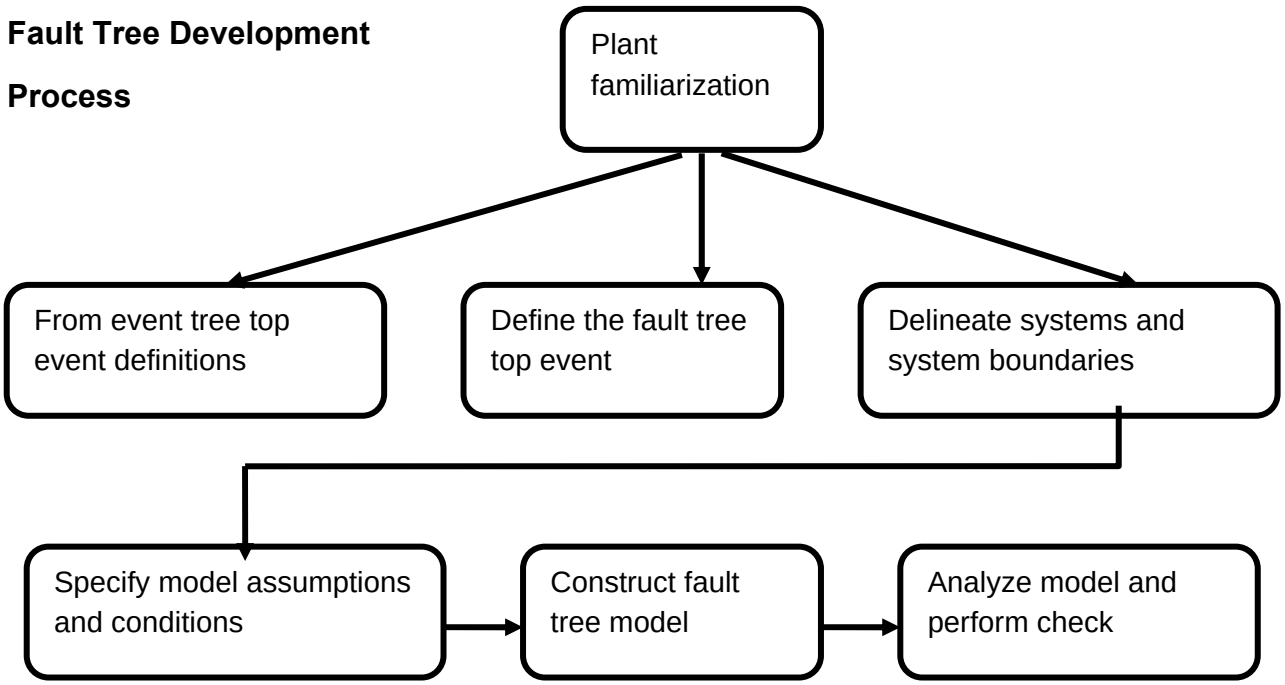
Section 4 introduces fault tree development, SAPHIRE fault tree symbols, and SAPHIRE fault tree modeling conventions. You will learn how to enter and edit fault trees by using the fault tree graphical editor and the graphical fault tree logic (i.e., textual) editor.



4.1. Fault Tree Development

Definition: Fault tree analysis is a deductive failure analysis method which focuses on identifying all of the credible ways that can cause an undesired event to occur. The undesired event is stated at the top of the fault tree. The fault tree gates specify the logical combinations of basic events that lead to the top event.

Fault Tree Development Process



4.2. Basic Event Symbols

Graphical toolbar symbols for different basic event types

Basic Event 

Boxed Basic Event 

Table of Basic Events 

Undeveloped Event 

House Event 

Undeveloped Transfer 

Fault Tree Graphic Symbols for Basic Events

Basic Event

A basic event represents a fault such as a hardware failure, human error, or an adverse condition. The circle signifies that the fault event does not require further development.

Boxed Basic Event

An alternate symbol for a basic event is a boxed basic event that provides a box to contain the description of the basic event. This shape is the standard basic event one normally finds on PRA fault trees.

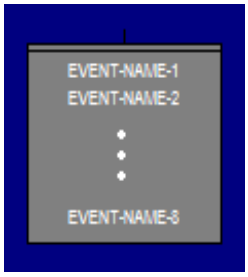


Table of Basic Events

The table of basic events symbol allows up to 8 basic events to be entered in a space-saving layout. The logic used by the table is dictated by the **gate** it is connected to in the fault tree. The name for the table must be specified but does not appear on the fault tree graphic.

Undeveloped Event

The undeveloped event denotes a basic event that is actually a more complex event that has not been further developed by fault tree logic. SAPHIRE treats this event no differently than a basic event.

House Event

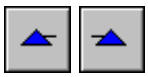
The house event denotes a failure that is guaranteed to occur (TRUE) or never to occur (FALSE). However, the calculation type assigned to a basic event establishes whether or not an event is a house event. Consequently, any basic event in SAPHIRE can be a house event, but the calculation type dictates the analysis behavior (see Section 5).

Undeveloped Transfer

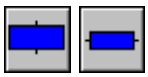
The undeveloped transfer indicates that the event is complex enough to have its own fault tree logic developed elsewhere; however, the event has been treated as a basic event in the present fault tree.

Graphical toolbar symbols for different graphic objects

Right and Left Transfer



Vertical and Horizontal Boxes



Connecting Lines

These shapes may be drawn on the fault tree, but are neither a basic event nor a gate. Instead, they are just graphical symbols. For example, a vertical box allows further descriptive information to be placed in the diagram, but SAPHIRE ignores this box when processing the fault tree.

4.3. Logic Gate Symbols

Graphical toolbar symbols for different gate types

AND Gate 

OR Gate 

N/M Gate 

Transfer Gate 

NOT AND (NAND) Gate 

NOT OR (NOR) Gate 

Inhibit Gate 

Fault Tree Graphic Symbols for Gates

Gate

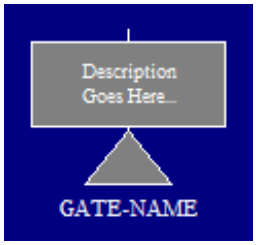
All inputs to the AND gate must occur for failure to occur. Text symbols used to represent an AND gate include ∩ and *.

OR Gate

Any one input to the OR gate will cause failure to occur. Text symbols used to represent an OR gate include ∪ and +.

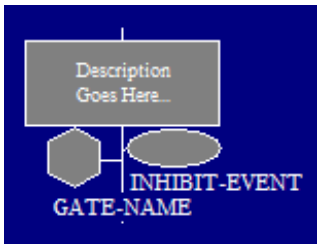
N/M Gate

This gate states that N of the M input events must occur for failure to occur. For a 2-of-3 gate, any combination of 2 of the 3 input events must occur.



Transfer Gate

This gate is used to link logic structures together without introducing any new logic of its own. The transfer gate indicates that logic is continued on a new page (or on the same page). The transfer gate **name** is the same as the gate name where the logic continues. When transferring to another page (a separate fault tree file), the gate being transferred to must be the top gate on the page. (Note that a top gate name must be the same as the fault tree name. Thus, the transfer gate name must be the same as the fault tree name.)

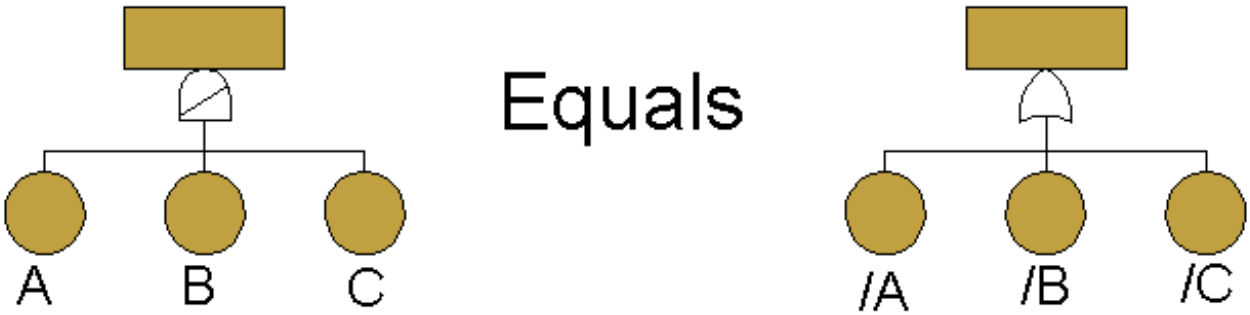


Inhibit Gate

The output occurs if the input occurs in the presence of an enabling condition (indicated by INHIBIT-EVENT). Thus, the inhibit gate is a special type of AND gate. The enabling condition (or conditioning event) is simply a basic event.

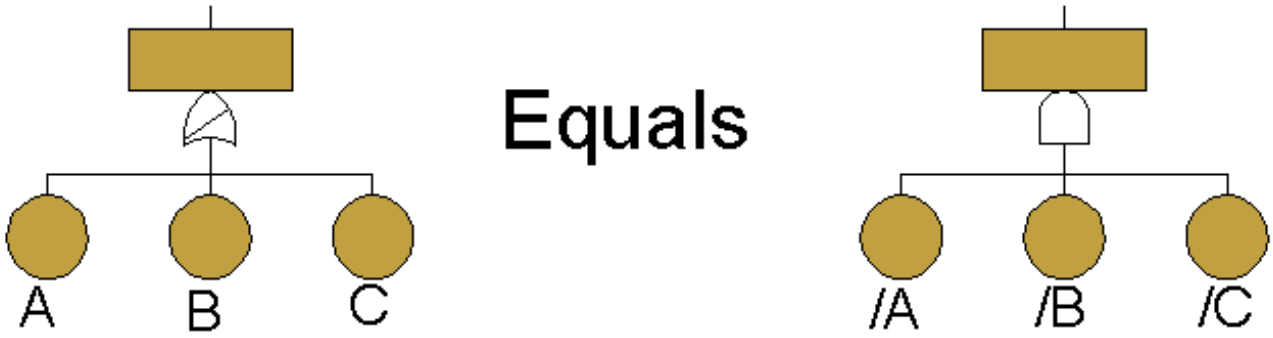
NOT AND (NAND) Gate

The output occurs if **any one** of the inputs **does not** occur (see example below). A NAND gate is converted into regular (non-complemented) logic by first complementing each input (A --> /A) and then changing the gate to an OR gate. Note in SAPHIRE that the "/" indicates a complemented event.



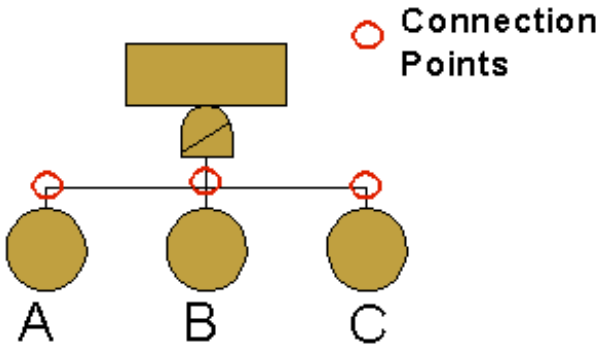
NOT OR (NOR) Gate

The output occurs if **none** of the inputs occur (see example below). A NOR gate is converted into regular (non-complemented) logic by first complementing each input (A --> /A) and then changing the gate to an AND gate. Note in SAPHIRE that the "/" indicates a complemented event.



Connecting Lines

Connecting lines can be solid or dashed, or a dotted/dashed line, but the standard is solid lines. Connecting lines can be drawn at any angle, but the lines must touch the symbols input or output stem to physically link the symbol.



4.4. SAPHIRE Fault Tree Conventions

Fault Tree File Name
The fault tree name **must** be the same as the top gate name. The fault tree name can be 24 characters long.

Fault Tree Gates
24 character names allowed. Gates may be duplicated in the tree or among different fault trees as long as the gates are identical. Identical gates are two gates with (1) the same name and (2) the same inputs.

Basic Event Names
24 characters allowed.

Top Gates
A fault tree "page" or file can have only one top gate.

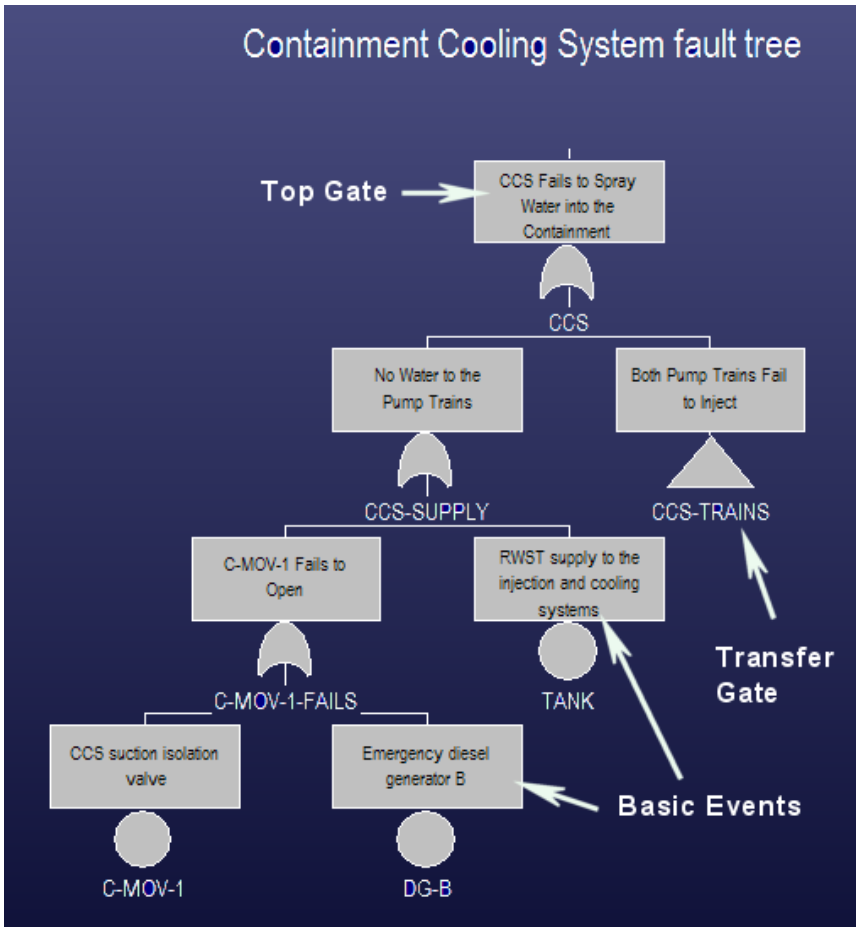
SAPHIRE Default Naming
SAPHIRE will automatically assign basic event names (EVENT-page#-event#) and gate names (GATE-page#-event#); however, the user may replace the default name. These defaults may be modified on the *Preferences* dialog (**View ➔ Preferences**).

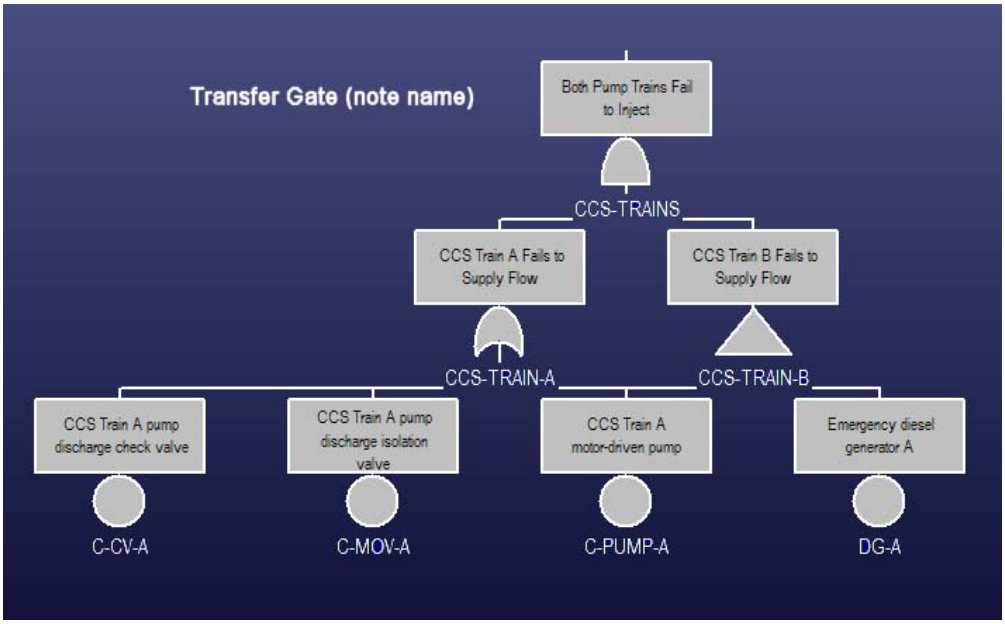
Transfer Fault Trees
A fault tree can transfer to another fault tree by using transfer gates. The transfer gate name and the "transfer to" gate name must be the same. The gate being transferred to must either (1) be on the same page or (2) be the top gate of a separate fault tree file.

Complemented Events
Complemented events can be input into fault trees by entering the basic event name beginning with the "/" symbol, e.g., /DG-A.

- Do not use ***, ?, \, @, /, or space** in the naming SAPHIRE fault trees or basic events. Note that the "/" symbol is reserved to denote a complemented event.

Example Fault Trees



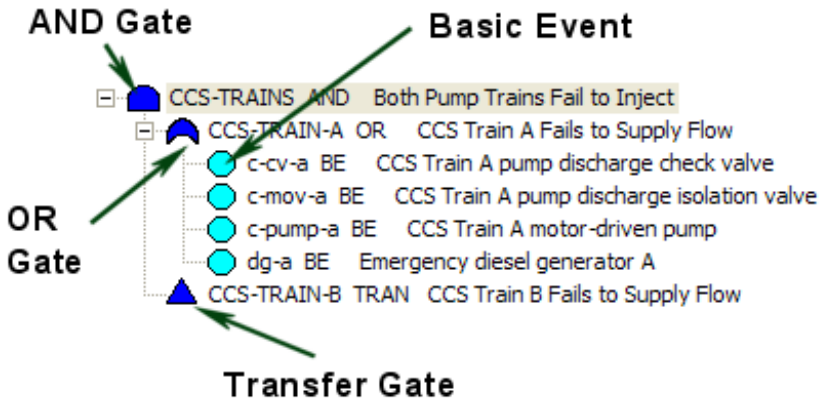


4.5. Fault Tree Logic Editor

The SAPHIRE fault tree logic editor allows you to construct the logic of a SAPHIRE fault tree (system or subsystem). Fault tree gate descriptions can be entered from this editor as well as basic event descriptions and data. The fault tree logic editor is a tool for editing fault tree logic, including gate and basic event data, in a textual, hierarchical manner.

Shown below is the logical representation of the CCS-TRAINS subtree shown on the previous page, but in the logic format.

Logic editor format



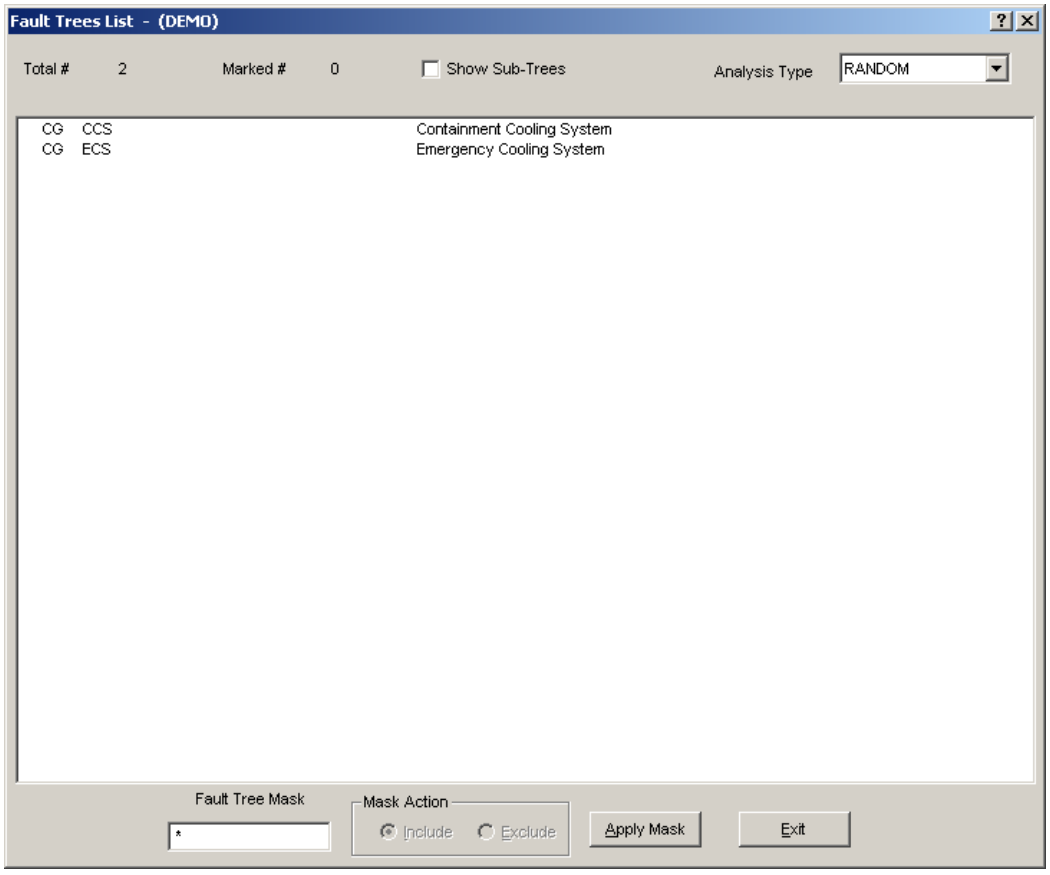
Line Format: Symbol, Name, Type, Description

The Fault Tree Logic Editor

- Select the **Fault Tree** menu.

- When you select this menu, the *Fault Tree List* dialog is displayed.

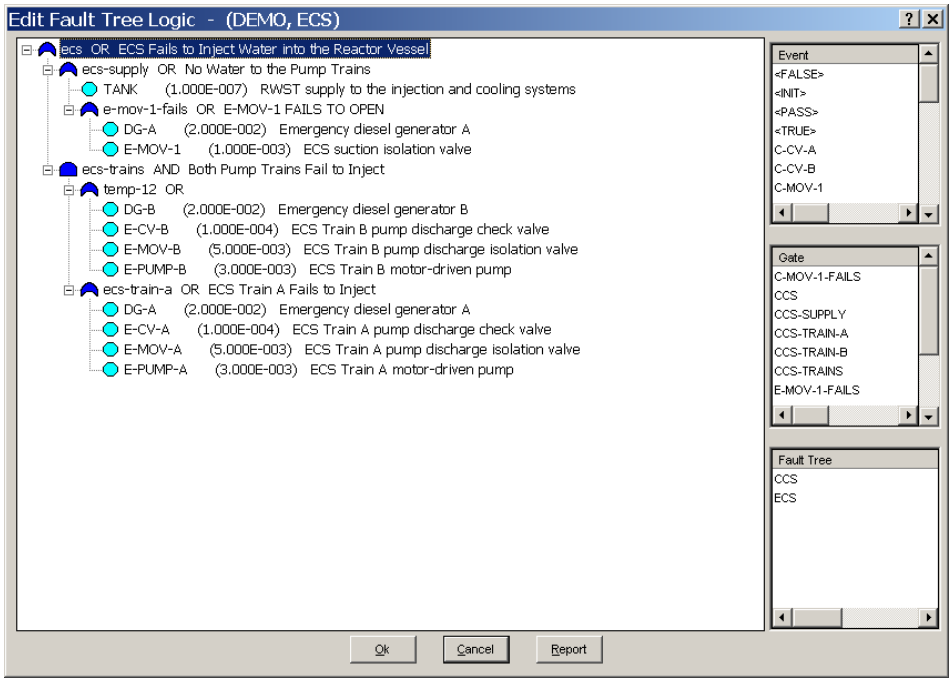
  The fault trees are shown in the list box. To display both systems *and subsystems*, select the Sub-systems check box. (Deselecting the Sub-systems check box will toggle the list so that only systems are displayed.)



To edit a fault tree, highlight the fault tree and select Edit Logic from the pop-up menu. (Right-click to invoke the pop-up menu.)

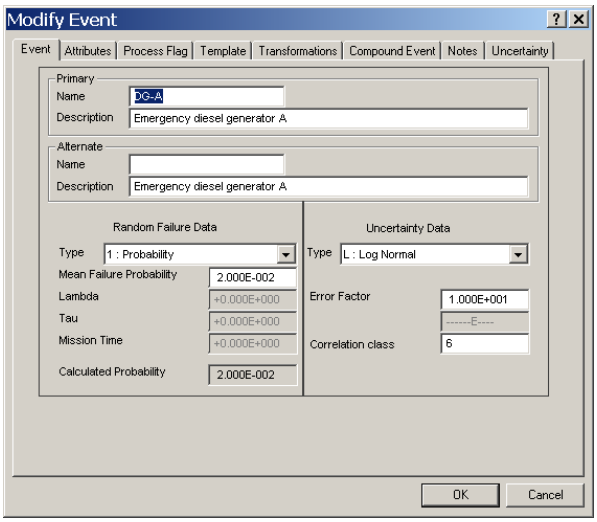
Fault trees are added to the database by adding tops to an event tree, by using **Modify ➔ Fault Tree ➔ Add**, or by **Fault Tree ➔ Add Fault Tree**.

The Logic Editor Display Screen

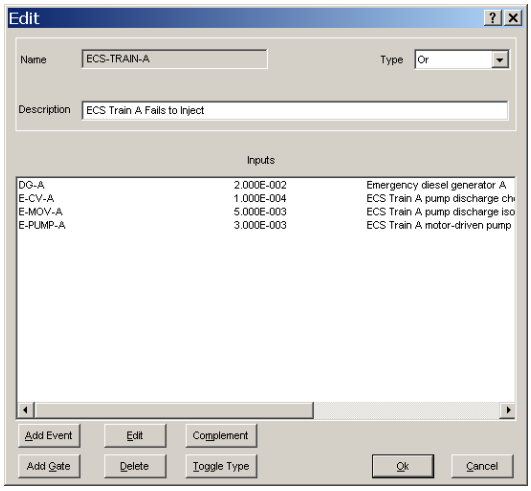


- The logic editor uses a hierarchical approach to editing the fault tree.
- The logic can be expanded by pressing + or collapsed by pressing -.
- To modify a basic event (or gate), highlight the item, click the right mouse button, and select **Modify**

For a basic event:



For a gate:



- Highlight the basic event (or gate) and select **Edit**. Then modify its name, description, or probability.

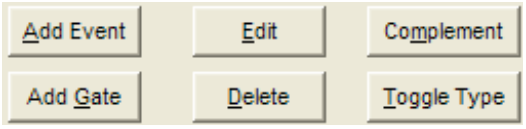
- To add a gate (or basic event), highlight the gate it inputs to (click the right mouse) and select modify.

- Select the **Add Gate** option and type in the name of the new gate along with its type and description.

- To add a basic event (or gate), highlight the item where it will be input, highlight the item from the list to the right of the window (or add item to list by right click and **Add**), and then right click and **Select** or just double click the item.

Logic Editor Modify Options

The following provides a list of the options available for adding or modifying both basic events or gates (from the Edit window).



Add Event – Add event is used to add a new basic event as an input to the gate. This option allows for the basic event's name and description to be added.

Add Gate – Add gate is used to add a new gate as an input to the gate highlighted. This option also allows for the gate's description to be added.

Edit – The edit option allows you to change a basic events name, description, and probability and change a gate's name, type, or description depending on which type of event is highlighted.

Delete – The delete option will delete the highlighted basic event or gate.

Complement – The complement option will complement any basic event that is highlighted.

Toggle Type – Toggle type will toggle the highlighted object between a basic event or a gate.

Ok – Close the Modify (Edit) fault tree dialog and apply the changes.

Cancel – Close the Modify (Edit) fault tree dialog without applying the changes.

Other Features of the Logic Editor (via the right mouse click)

Find Option

- The find option searches the fault tree logic to find the specified basic event or gate.
- To find a basic event or gate (right click the mouse) and select **Find**. Type in the name of the gate or basic event and check the box if the search is for a gate and uncheck the box if the search is for a basic event. Press **Find**.

Move Option

- The move option will move a gate and its logic from its original input gate to another selected gate.
- To move a gate, highlight the gate (right click the mouse), select move and place cursor by the new gate and click the left mouse button. The gate and its associated logic will now be placed as an input to the new gate.

Transfer Option

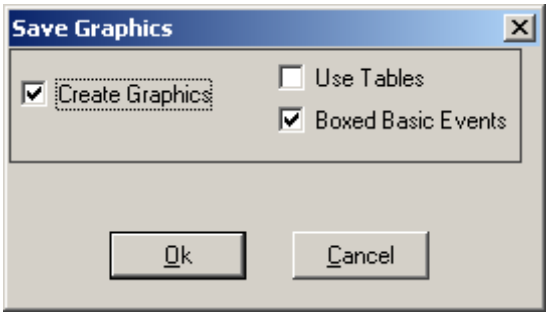
- The transfer option transfers from the fault tree currently being edited to the fault tree specified by the transfer gate.
- To transfer from one fault tree to another highlight the transfer gate (right click the mouse) then select **Follow Transfer**.
- To transfer back, highlight the transfer gate (right click the mouse) then select **Transfer Back**.

4.6. Updating the Graphical Fault Tree (.DLS) File

After changes are made to fault tree logic using the Logic Editor, when you press **Ok** a dialog box will ask if you what to save the fault tree logic as a graphic file.

At this point you can check the "Create Graphics" box and a fault tree graphics file will automatically be created.

Alternatively, at any time, one can use the **Utility ➔ Fault Tree ➔ Alpha-to-Graphics** conversion to make an updated fault tree graphical file. (You do not have to update the graphical file in order to perform analyses with the updated logic; however, to keep the fault tree logic and the fault tree graphic identical, you must perform the Alpha-to-Graphics conversion if you change the logic but not the graphic.)



4.7. Fault Tree Graphics Features Guide

This section provides an overview of a variety of activities one may wish to perform when creating and editing fault trees using the **graphical** editor. Included in this section is information on the following:

Begin a fault tree graphic editing session

To develop a new fault tree

SAPHIRE menu bar: **Fault Tree**

Fault Tree List dialog pop-up menu: **Edit Graphics**

Fault Tree Editor menu bar: **File ➔ New**

or

SAPHIRE menu bar: **Fault Tree**

Fault Tree List dialog pop-up menu: Add Fault Trees

If you wish to edit an existing file while in the graphic editor

Fault Tree Editor menu bar: **File ➔ Open**

OR Choose the **Open File** button on the tool bar.

Double-click on the fault tree filename you wish to edit.

Select or arrange logic symbols and basic events

To begin building a fault tree

Select the desired object button from the tool bar (left click).

Then move the shape cursor to the desired location with the mouse and click the left mouse button to “drop”: the shape. This process can be repeated until the right mouse button is clicked.

Use the right mouse button to end the selection.

To select objects

Choose the Pick button to select a single object

Choose the Text Pick button to select a text object. 

Then click on the desired object.

A dashed line will appear surrounding the selected object(s).

To delete unwanted objects

Select the object to be deleted (click on the shape with the left mouse button)

Fault Tree Editor menu bar: **Edit ➔ Delete**

OR press the Delete key.

To move symbols (and their associated text, if any)

Select the object to be moved (individually or by drawing a select "region" around several objects).

Drag the object to the desired location.

Name gates and basic events

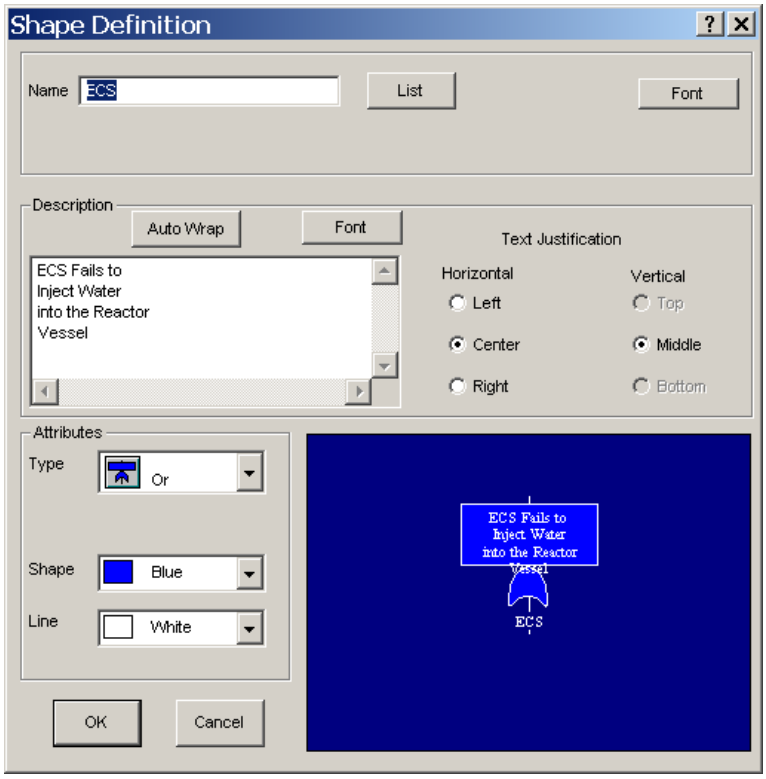
To rename logic or basic event symbols

Select the desired shape

Fault Tree Editor menu bar: **Edit ➔ Attributes**

OR right-click and choose **Edit**,

Then make desired changes.



To view the current name of a gate or basic event

Choose the Pick button to select a single object. On the bottom of the fault tree graphic the basic event name or gate name along with their description will be shown.

Connect symbols together with lines

To link gates and basic events together

Select the **Line** button from the tool bar. Drag from the starting location to the ending location.

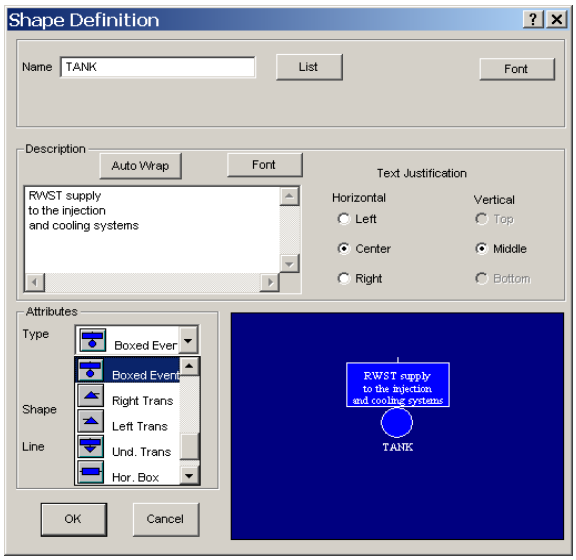
(Note: if you move symbols after connecting with lines, you may need to redraw the lines unless the lines were moved with the symbols.)

Change a symbol type

To change a symbol (e.g., to change a basic event to an undeveloped transfer)

Select the shape

Fault Tree Editor menu bar: **Edit ➔ Attributes**



Choose the **Type** button.

Select the new shape type from the list.

Modify the screen view

To change the view of the fault tree

Fault Tree Editor menu bar: **View**

Page Up - allows you to move up one page.

Page Down - allows you to move down one page.

Page Left - allows you to move left one page.

Page Right - allows you to move right one page.

Zoom - allows you to zoom in or out. Or select the button 

View ➔ Normal resets the screen.

View ➔ Fit zooms the fault tree in to fit the screen. Or select the "fit to page" button

Or

Highlight an area right mouse click and select **zoom selected**.

To redraw the window

Fault Tree Editor menu bar: **View ➔ Refresh** or press the **F5** key.

Make a text box on the graphic

To enter text (for a title or other description)

Choose the Text Object button from the tool bar. The cursor will change to the Text cursor.

Position the text cursor at the desired location.

Click the mouse. The *Text Attributes* dialog will be displayed.

To move the associated symbol only

Choose the Pick button from the tool bar. The cursor will change to the Pick cursor.

To move text only

Choose the Text Pick button from the tool bar. The cursor will change to the Text Pick cursor.

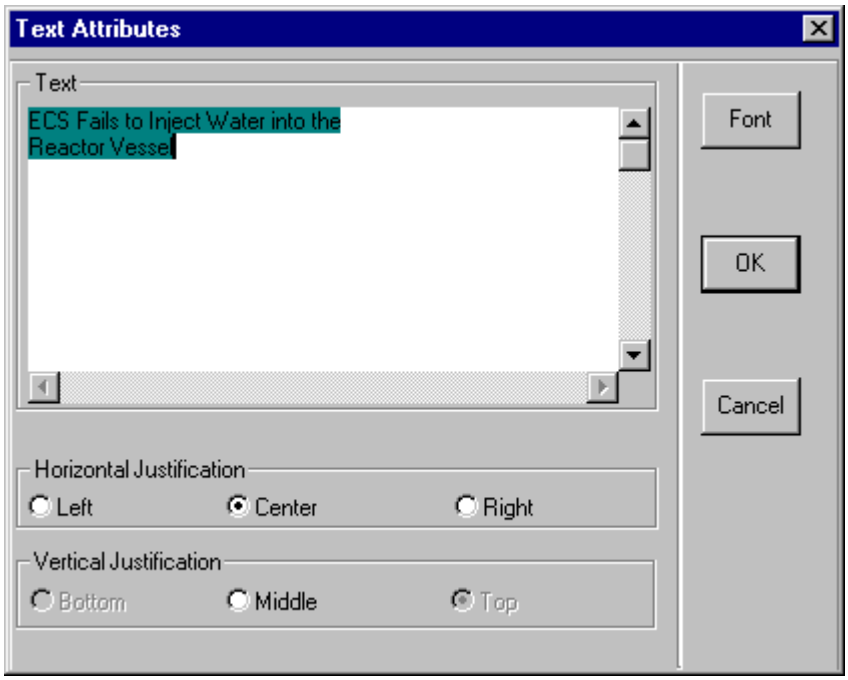
Select the desired text.

Drag the selected text to the new location.

To change existing text and attributes

Select the desired text using the Text Pick cursor.

Right-click and choose **Edit**. The *Text Attributes* dialog will be displayed.



Text - Descriptive text for a shape or explanatory text for the entire fault tree. Maximum of 600 alphanumeric upper- and lowercase characters.

Justification

Horizontal - The horizontal alignment of the text.

Vertical - The vertical alignment of the text.

Text Font - The font size or type for the text and the color of the text.

OK - Close the *Text Attributes* dialog and add the input text to the diagram.

View the graphical editor preferences

Fault Tree Editor menu bar: **View➔ Preferences**

OR right-click and choose **Preferences**

Note the **Snap Line to Shape** option - If checked, lines drawn from any part of one shape to any part of another will be adjusted to connect the two shapes. If not checked, lines will be placed exactly where dropped.

Check boxes under **Other** to do the following:

Show Names: This places the event Primary name under the symbol.

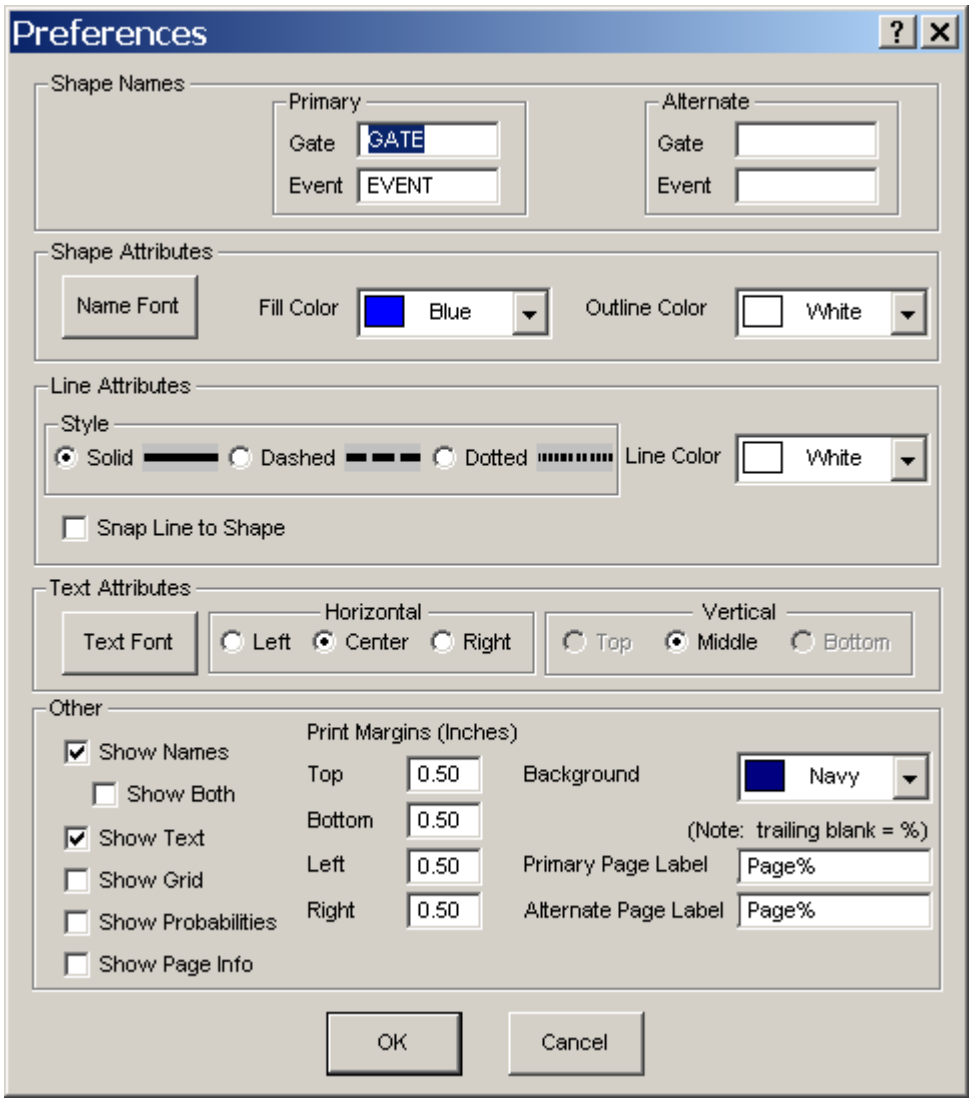
Show Both: This places the event Primary and Alternate name under the symbol.

Show Text: This will show the event or fault tree text.

Show Grid: This will show grid lines to help in placing symbols.

Show Probabilities: This will put the basic event's probability to the side of the event.

Show Page Info: This will place the fault tree title and the date the fault tree was last modified along with the page number for the fault tree on the bottom of the graphic.



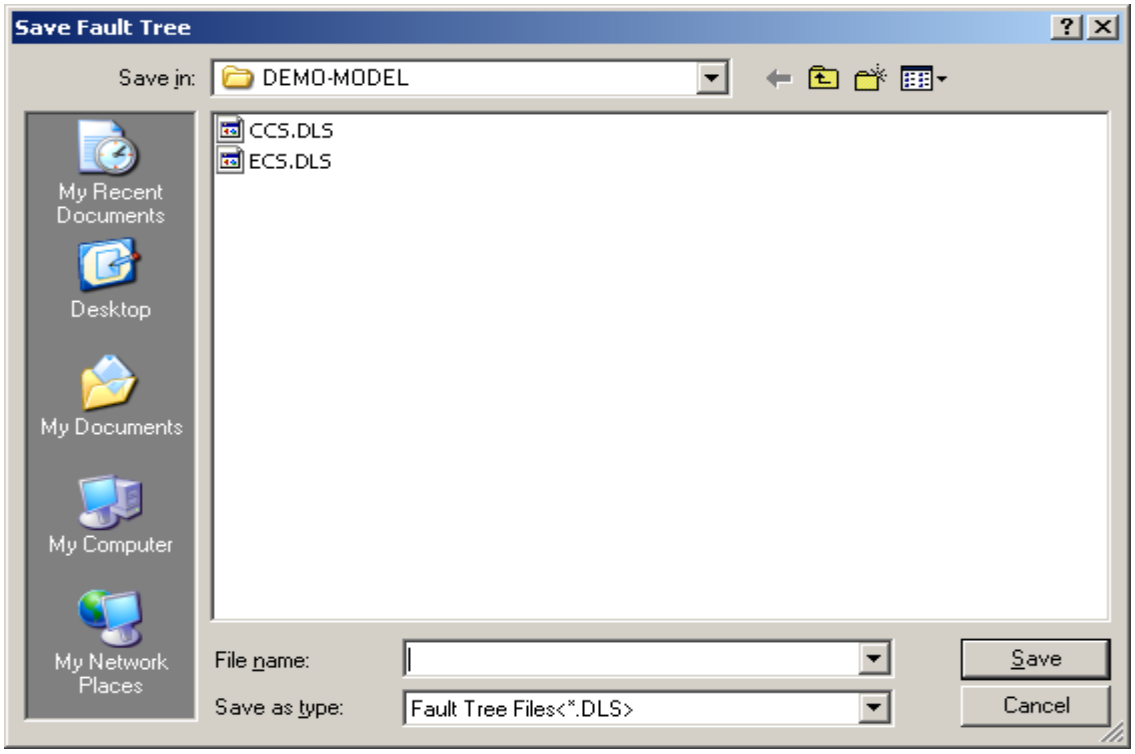
End the fault tree editing session

To save the fault tree graphic

Fault Tree Editor menu bar: **File ➔ Save**,

OR Choose the **Save Diagram** button on the tool bar 

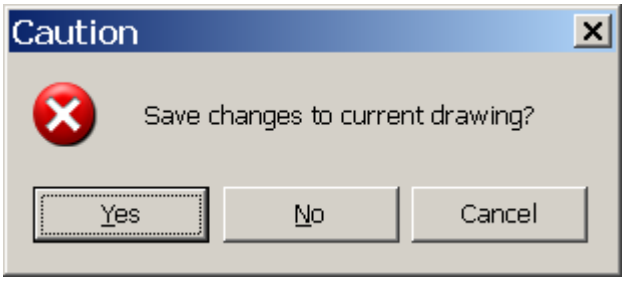
Name the fault tree file as directed.



To exit without saving

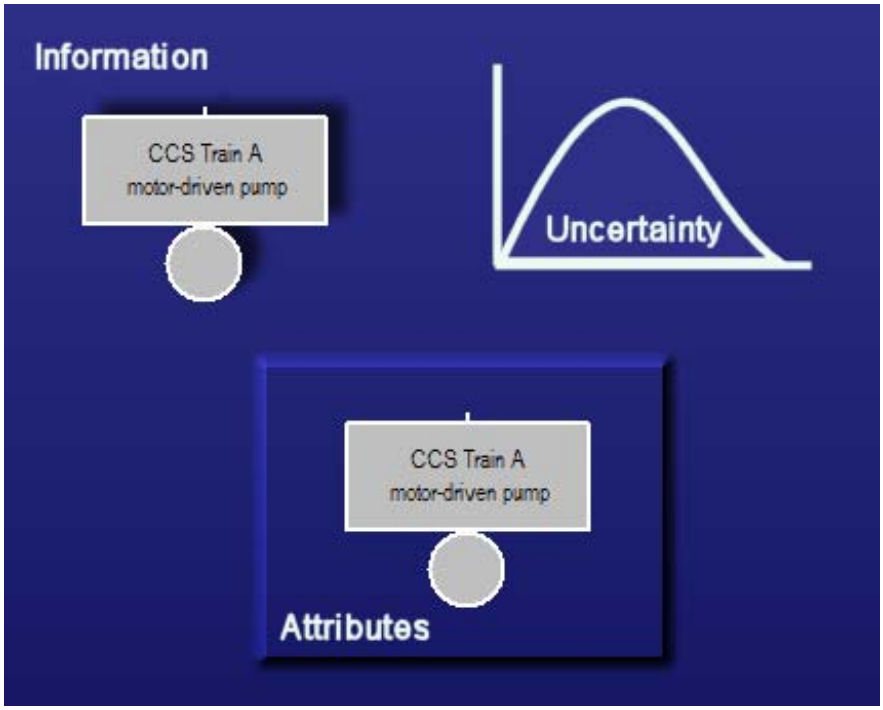
Fault Tree Editor menu bar: **File ➔ Exit**

Choose **No** to quit without saving.



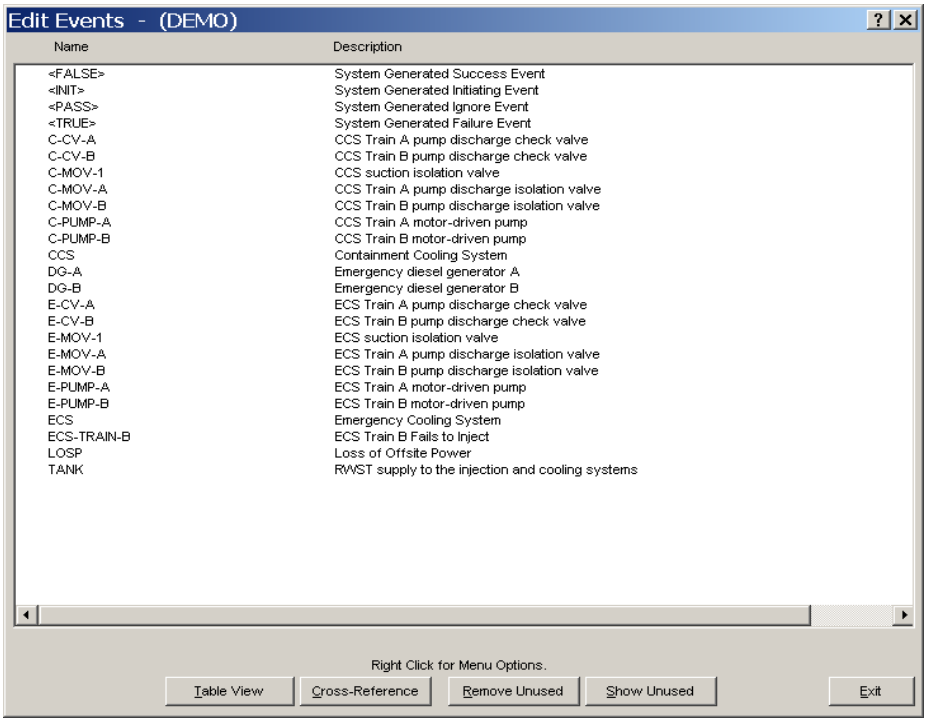
5 BASIC EVENT INFORMATION

Section 5 introduces SAPHIRE basic event data entry and basic event probability calculation types.



5.1. Modify Basic Events

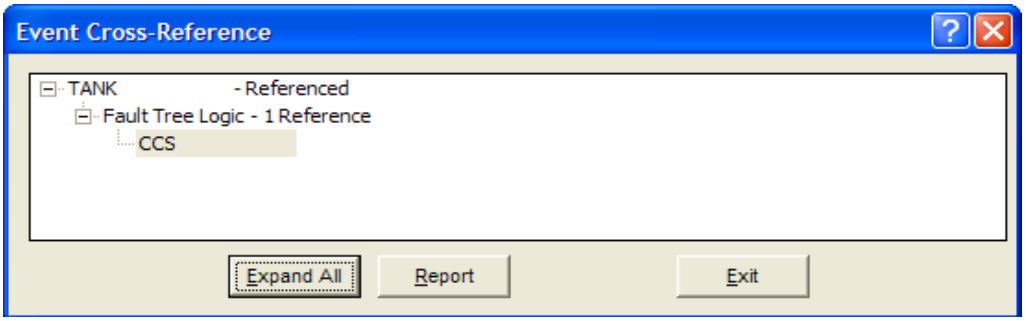
- To enter basic event data, select **Modify** from the menu. Then select **Basic Events**.



- To modify data for an existing event, double-click on the event you want to edit or right-click to invoke the pop-up menu and select **Modify**.

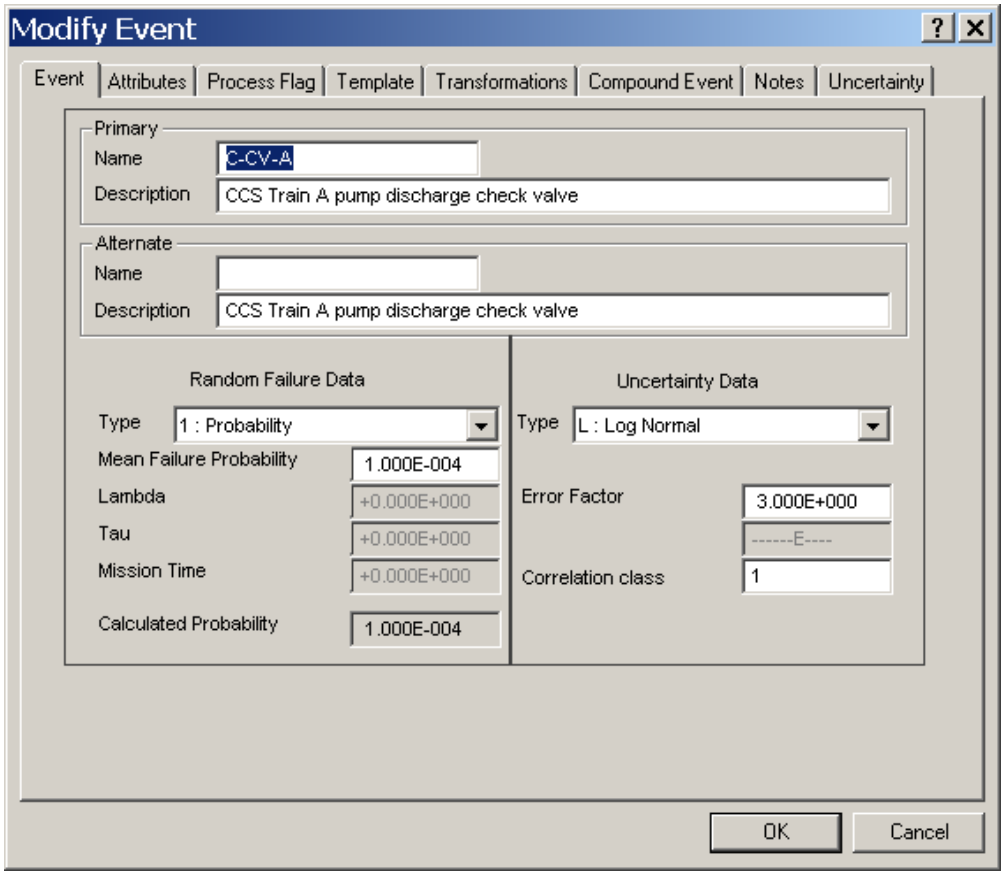
- To add a new event, right-click to invoke the pop-up menu and select the **Add** option.

- The “Cross-Reference” option allows one to view the places in the project where a basic event might be used. For example, if a basic event is used in a fault tree, the cross-reference will indicate which tree by name.



5.2. Basic Event Data Fields

Modifying a basic event displays the “Modify Event” dialog.



Primary Name

The Primary name is the fundamental name used in the fault trees and event trees. A unique Primary name must be specified for every basic event in the logic models. A maximum of 24 uppercase, alphanumeric characters may be entered. Embedded spaces **are not** allowed.

Alternate Name

The Alternate name, which can be different than the Primary name, can be used to report cut set results by selecting Alternate name in the **Define Constants** option. This feature allows cut sets to be reported using a different naming scheme. A maximum of 24 uppercase, alphanumeric characters may be entered in this field. Embedded spaces are not allowed.

Primary Description

This is a 120-character, uppercase or lowercase, alphanumeric field that provides brief, descriptive information.

Alternate Description

This is a 120-character, uppercase or lowercase, alphanumeric field that provides brief, descriptive information.

Type – Calculation Type

In the Failure Data section the calculation type is a numerical reference to the calculation method to be used. There are 14 unique calculation types identified as 1, 3, 5, 7, V, T, F, I, C, X, S, G, E, and H. Choose the desired calculation type from the drop-down list.

The "basic" calculation types are:

Calc Type	Equation	Comments
1	$P = p$	Simply a probability (or, possibly, a frequency in the case of an initiating event). Could represent a failure upon demand. p = probability, in the "mean probability" field
3	$P = 1 - EXP(- \lambda T_m)$	Failure probability of an operating component without repair (non-demand failure mode) λ = mean failure rate, in the "lambda" field T_m = mission time, in "mission time" field
5	$P = ([\lambda \tau] / [1 + \{\lambda \tau\}]) * (1 - EXP [-(\lambda + 1/\tau) * T_m])$	Failure probability of an operating component with the possibility of repair following a failure. λ = mean failure rate, in the "lambda" field τ = mean repair time, in the "tau" field T_m = mission time, in "mission time" field
7	$P = 1 + (EXP[-\lambda \tau] - 1)/(\lambda \tau)$	Failure probability of a standby component in a non-demand failure mode that is tested periodically. λ = mean failure rate, in the "lambda" field τ = test interval, in the "tau" field

Calc Type	Equation	Comments
T	P = 1	House event TRUE, represents a failure.
F	P = 0	House event FALSE, represents a success.

The "advanced" calculation types are:

Calc Type	Equation	Comments
I	P = 0	The basic event is to be treated as if it did not exist in the logic for the fault tree. Before the tree is solved, all references to the specified event are removed from the fault tree.
V	P = V	The basic event is to be treated as a value event. This calculation type is used in order to include events that take on values instead of probabilities. Therefore, the value can be greater than 1.0.
C	P = calculated probability	The basic event's probability is calculated based on different equations within SAPHIRE (i.e., summation of two different basic events or common cause failure calculation performed within SAPHIRE).
X	P = calculated probability	This basic event is a human action and SAPHIRE has worksheets built in to calculate the human error probability based on performance shaping factors.
S	P = calculated probability	Finds a fault tree with the same name and uses its current cut set min cut upperbound probability as the basic event's probability.
G	$P = \Phi[\ln(g_{specified}/a)/B_r]$.	The basic event is to be treated as a seismic event. The probability value for screening will be calculated using the "screening G-level" (ground acceleration) and median failure acceleration (fragility) specified by the user. Strictly for use only in seismic analyses.

Calc Type	Equation	Comments
H	$P = \Phi[\ln(g_{hazard}/a)/B_r]$.	The basic event is to be treated as a seismic event. The probability value for screening will be calculated using the highest G-level (ground acceleration) from the project's "seismic hazard curve" and median failure acceleration (fragility) specified by the user. Strictly for use only in seismic analyses.
E	P = calculated probability	Finds an end state with the same name and uses its cut set min cut upperbound probability as the basic event's probability

Notes: Φ = standard normal cumulative distribution function
a = median failure acceleration (the approximate ground acceleration sufficient to cause the component to fail)
B_r = amount the failure acceleration "a" can vary
g = ground acceleration for screening

Mission Time

The mission time is the period of time that a component is required to operate in order to characterize the component operation as being successful. An example would be for a pump that must run for 24 hours after an initiating event, the mission time would be 24 hours.

Distribution Types

For the Uncertainty Data section, there are thirteen predefined distribution types available. The predefined distribution types are:

> Normal, lognormal, beta, dirichlet, gamma, chi-squared, exponential, uniform, maximum entrophy, constrained non-informative, histogram, triangular, and seismic.

In addition to these predefined distribution types, user-defined histograms may be used.

The default distribution type is no distribution (i.e., it is a point estimate). Choose the desired distribution type from the drop-down list.

Value 1
Enter the first parameter of the distribution, if one is required.

Value 2
Enter the second parameter of the distribution, if one is required.

Correlation Class
Use to account for data dependencies among like events in the database. Enter up to four uppercase, alphanumeric values. A blank correlation class indicates that there are no data dependencies. When running the uncertainty analyses, the same sample value will be used for all basic events with the same correlation class.

Calculated Probability
The calculated probability that will be used for the basic event is listed in this field.

SAPHIRE Failure Data Entry Requirements

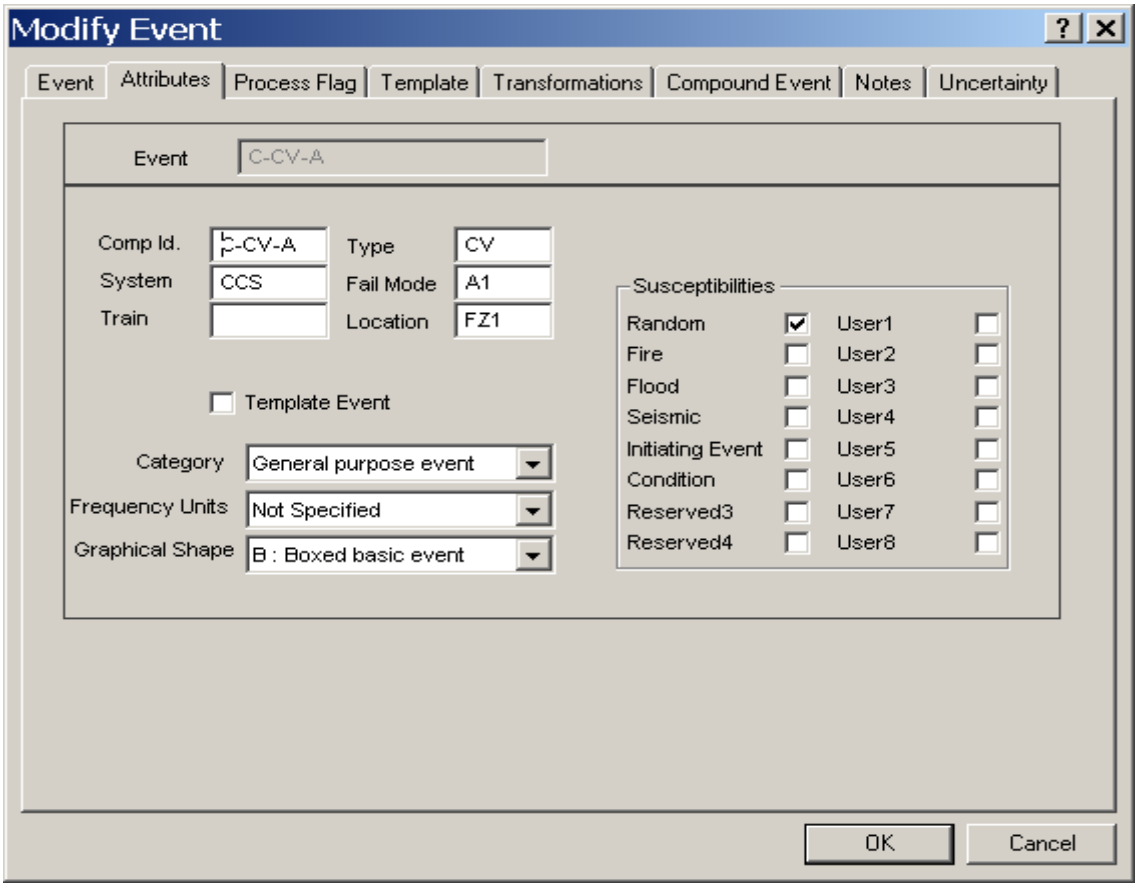
Calculation Type	Mean Failure Probability	Lambda* (per hour)	Tau (hours)	Mission Time (hours)
1	x			
3		x		x**
5		x	x	x**
7		x	x	

Notes: * The time units of lambda, tau, and mission time must be the same so that they cancel (e.g., in time units of either hour or per hour).

** If no mission time is specified (i.e., the mission time is zero), the default mission time specified in SAPHIRE **Utilities ➔ Define Constants** will be used.

5.3. Basic Event Attributes

Each basic event has special attributes that delineate the type of event.



Comp Id

Component Identifier. Enter up to seven (7) alphanumeric characters to identify a component by a unique designator. This is usually part of the component label (e.g., DG01). No embedded blanks are allowed.

System

Enter up to three alphanumeric characters to identify the system containing the component.

Train

Enter up to two alphanumeric characters to identify the train containing the component.

Type
Enter the event type attribute.

Failure Mode
Enter up to two alphanumeric characters to identify the failure mode for the component.

Location
Enter up to three alphanumeric characters to identify the physical location for the component.

Susceptibilities
The susceptibility flags indicate whether or not the event is susceptible to a specific kind of failure. There are 16 susceptibilities. Several are defined below:

1	=	Random (default)	2	=	Fire
3	=	Flood	4	=	Seismic
5	=	Initiating Event Assessment	6	=	Condition Assessment

Susceptibility flags must be checked to be considered susceptible to a specific type of failure. Note that all events are susceptible to random failure regardless of the random flag's value.

Category
Select from the drop-down list to specify the category or use of the event.

General purpose event - This is the default and is appropriate for must basic events.

‘I’ - Initiating event - Any initiating events should be identified with this category designation. The event tree editor will automatically enter an 'I' when the user specifies that the first event is an initiating event.

‘H’ - Hazard event - A special type assigned to histogram bins for seismic analysis.

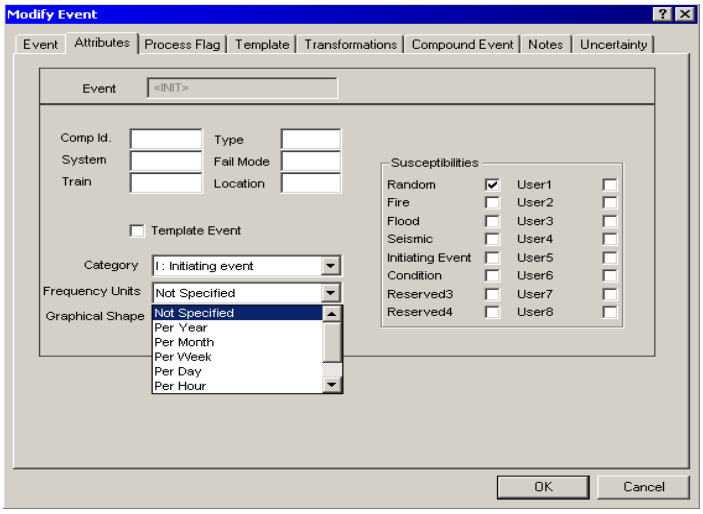
‘R’ - Recovery event - Events with this category designation will be listed in a list of recovery events used in the Recovery Rules editor.

‘S’ - Support System State – Reserved for future use.

'V' - Value event - Allows for values greater than one or less than zero to be used for the basic event. Represents a general "value" for an event.

Frequency Units

The frequency units can be specified for initiating events (e.g. per year). The default frequency units specified in **Modify ➔ Project** will override the frequency units specified if they are different. This is to ensure all sequences will be using the same frequency unit for summation.



Frequency units available are: Per Year, Per Month, Per Week, Per Day, Per Hour, Per Minute, Per Demand.

Graphical Shape

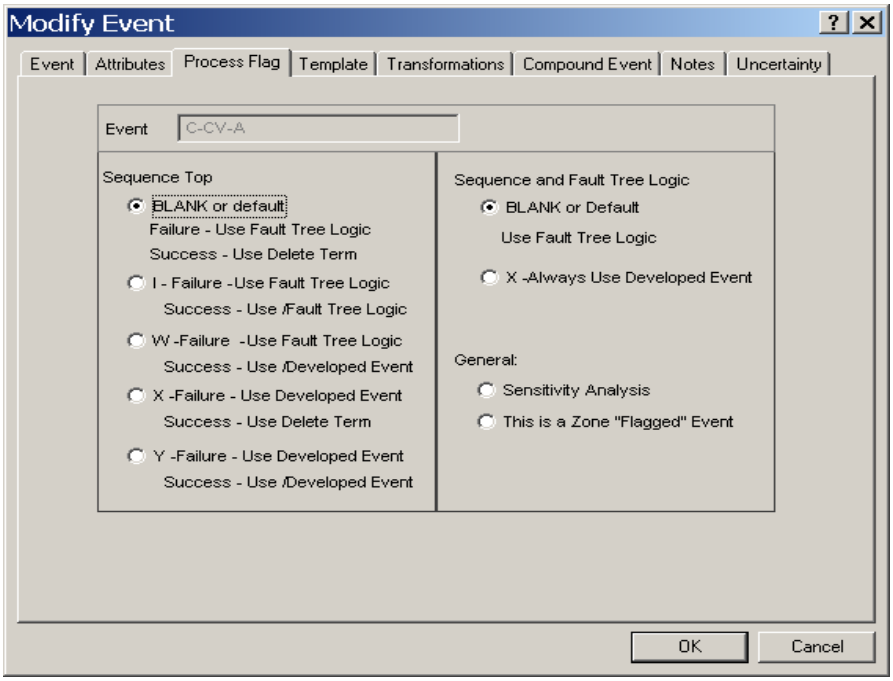
The graphical shape drop down list specifies the type of basic event symbol to use when an alpha-to-graphics is performed.

Template Event

'Template Event Check Box' – If checked, then the event's probability, attributes, and other information can be used by other basic events.

5.4. Basic Event Process Flags

Process flags are primarily used to tell SAPHIRE how to solve event tree accident sequences and fault tree logic.



Sequence Top Flags

BLANK or Default

When the Process Flag field is blank, the transfer associated with this event is expanded for failure references. For success branches in an event tree, the transfer is also expanded; however, the impossible cut sets are removed from the resulting cut sets using cut set matching (i.e., the delete term).

Process Flag = 'I'

Use system logic (if top event fails), use the complement of the system logic (if top event succeeds). That is, if the top event is a failure, SAPHIRE will expand the fault tree and solve, just as one expects. If the top event succeeds, SAPHIRE will complement the fault tree logic and solve it, thereby resulting in a non-coherent logic solution.

Process Flag 'W'

Use system logic (if top event fails), use complement of the developed event (if top event succeeds). That is, if the event fails SAPHIRE will expand the fault tree and solve. If the event succeeds, SAPHIRE will treat the top as a basic event (i.e., developed event) and use the complement of the event for the system probability.

Process Flag 'X'

Use developed event (if event fails), use cut set matching to eliminate cut sets (if event succeeds). That is, an "X" tells SAPHIRE that a basic event is to be used for failure probability, but a success top is to be treated the same as if the flag was blank.

Process Flag 'Y'
Use developed event (if event fails), use complement of developed event (if event succeeds). That is, a "Y" indicates that a transfer is to be replaced with its basic event for failed references and the complement of the event is to be used for success tops. If the top event is to be treated as a basic event (both for the up and down branch), then use a “Y” flag for the event.

Sequence and Fault Tree Logic Flags

BLANK or Default
When the Process Flag field is blank, SAPHIRE uses fault tree logic in the event tree top events during sequence cut set generation.

Process Flag ‘X’
When the Process Flag field is “X”, SAPHIRE uses a developed event instead of fault tree logic in the event tree top events during sequence cut set generation (used in conjunction with the calculation type “1” or “S” random failure data setting).

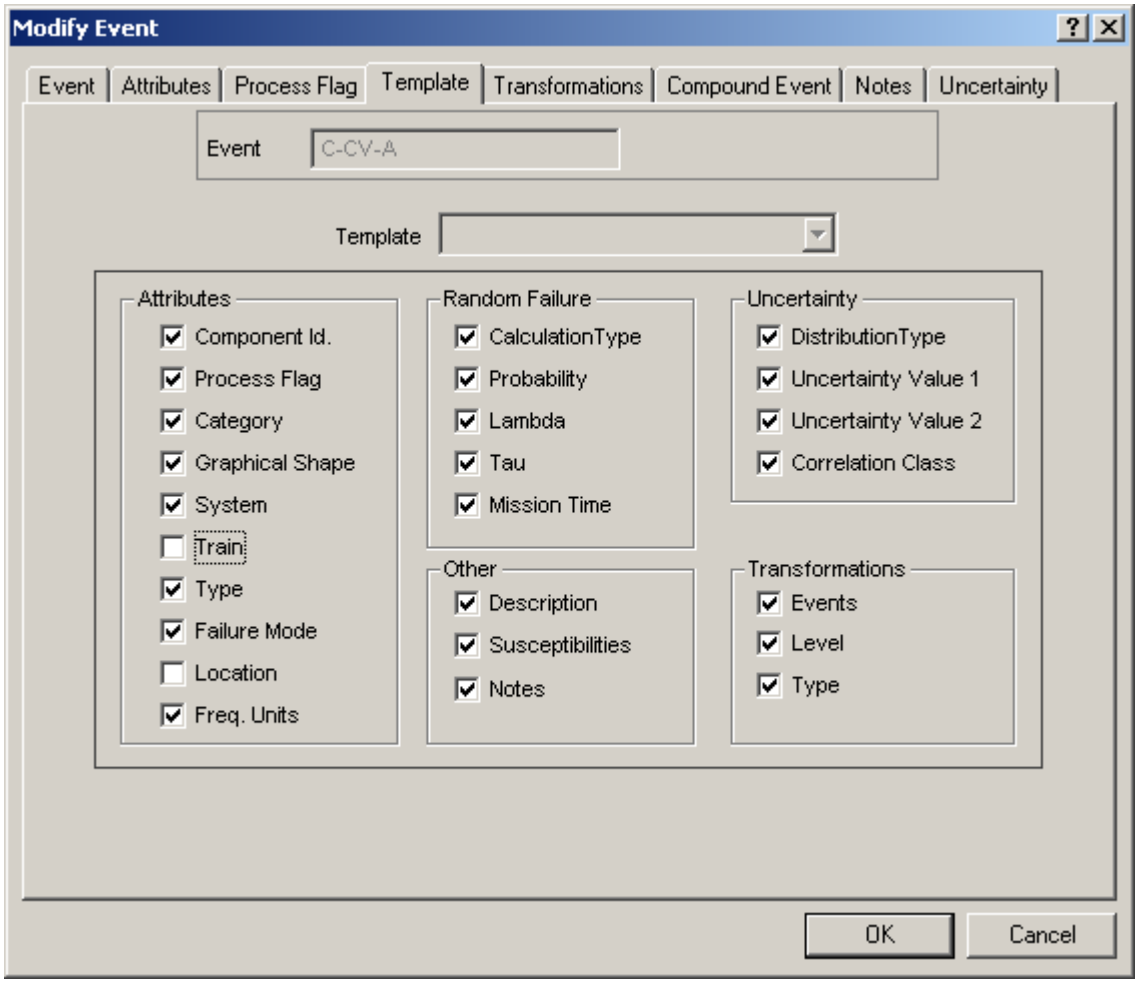
General Flags

Sensitivity Analysis
If an event is marked for sensitivity analysis, SAPHIRE will map a core damage frequency plot. A sensitivity analysis allows you to see how sensitive the core damage frequency is in relation to an event.

Zoned Flagged Event
A zone flagged event is an event that has been marked as representing a zone (i.e., location or area). An example of a zone is a fire zone or a flood zone. When SAPHIRE encounters a zone flagged event, it performs a transformation. A transformation is an event or set of events that replace a zone flagged event.

5.5. Basic Event Template

If one or more events have been identified as being “template” events (via the attributes tab), then the event may use the template information by selecting the template event.

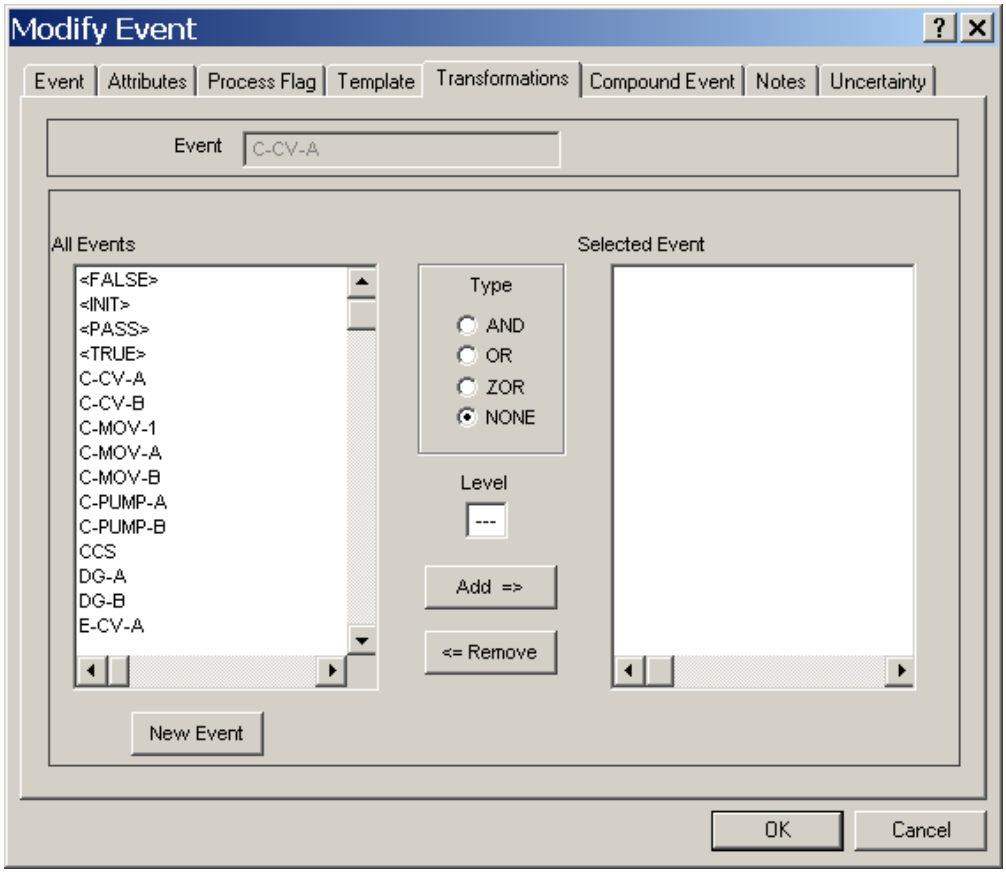


Template

If you wish to use another event’s information as a template for this basic event, select the name of the event from the drop-down list. Then check the box next to the desired characteristics to be used by this basic event. By default, all of the template event characteristics are selected.

5.6. Basic Event Transformations

Select basic event(s) from the *All Events* list and the desired **Type** and **Level**.



Transformation - Type
This field indicates the required behavior of the collective events. Enter one of the following:

AND = All included events must fail. Event is replaced with an AND gate, with all marked events as inputs.

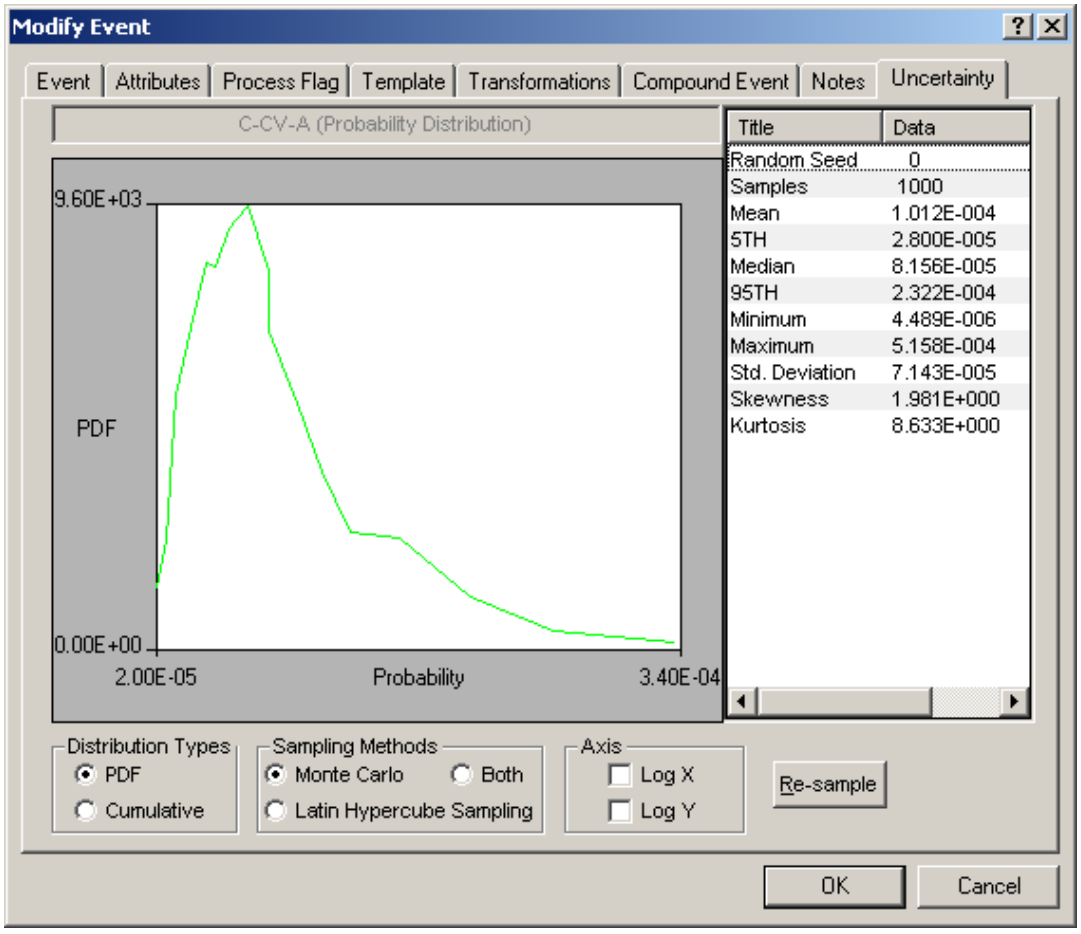
OR = Any included events must fail. Event is replaced with an OR gate with all marked events as inputs.

ZOR = Events make up the zone. If any events in the list fail, all events fail.

Transformation - Level
Enter an integer between 0 and 255 indicating the level of substitution for the transformation.

5.7. Basic Event Uncertainty

Summarizes the basic event uncertainty (if uncertainty parameters have been specified).



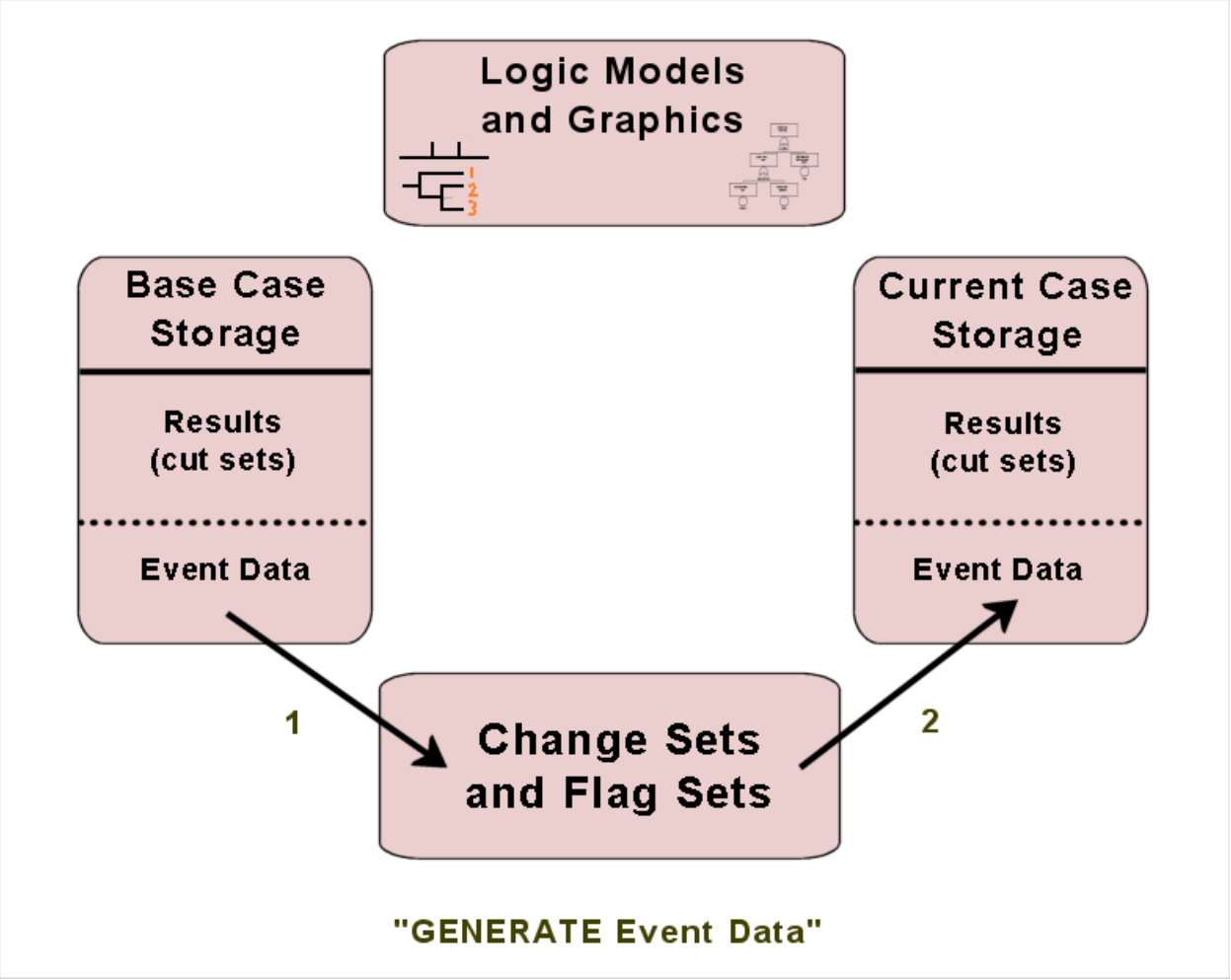
Terms You Should Know

Developed Event
A developed event in SAPHIRE is either an event tree top or the fault tree top gate. Regular fault tree basic event *are not* considered developed events.

Delete Term
In SAPHIRE, the process known as “delete term” refers to the removal of sequence success cut sets from the list of failure cut sets when generating sequence cut sets. As an example of the “delete term”, consider a sequence where top event A is successful and top event B is failed. Any cut sets that would fail A should not be allowed in the cut sets for B, so those cut sets are removed from the sequence.

5.8. Using "Generate" to Process Event Data

- Use the **Modify ➔ Basic Events** menu to enter basic event data.
- Return to the main SAPHIRE menu.
- Select **Generate** from the menu bar, and select the **Generate** button. To simply copy the base case data (the data just entered under the "Modify" option above) into the current case, do not mark any change sets.



6 GRAPHICAL OUTPUT

Section 6 describes the Alpha to Graphics conversion and Page Tree features for fault tree graphics, and how to print fault tree and event tree graphics.

6.1. The Fault Tree Alpha to Graphics Feature

Function — The "Alpha to Graphics" feature uses internally stored fault tree logic to create a graphical fault tree file that can be viewed or edited in **Fault Tree** menu and printed from the **Report ➔ Fault Tree ➔ Graphic** menus.

When to Use Alpha to Graphics

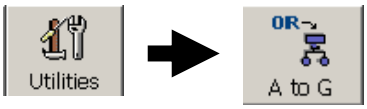
To replace the fault tree graphic that you created in **Fault Tree ➔ Edit Graphics** with a SAPHIRE-generated graphic that is evenly spaced and aligned or to use different basic event symbols.

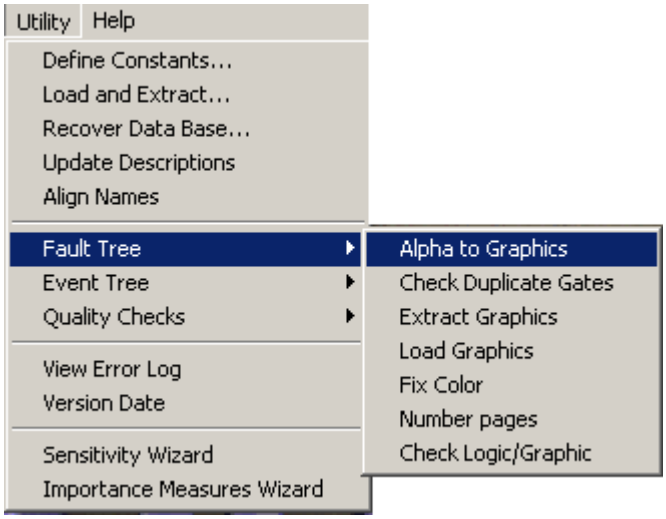
To update the existing fault tree graphic because you used the fault tree logic editor to modify fault tree logic.

To create a fault tree graphic after loading the fault tree logic into the database via the MAR-D interface.

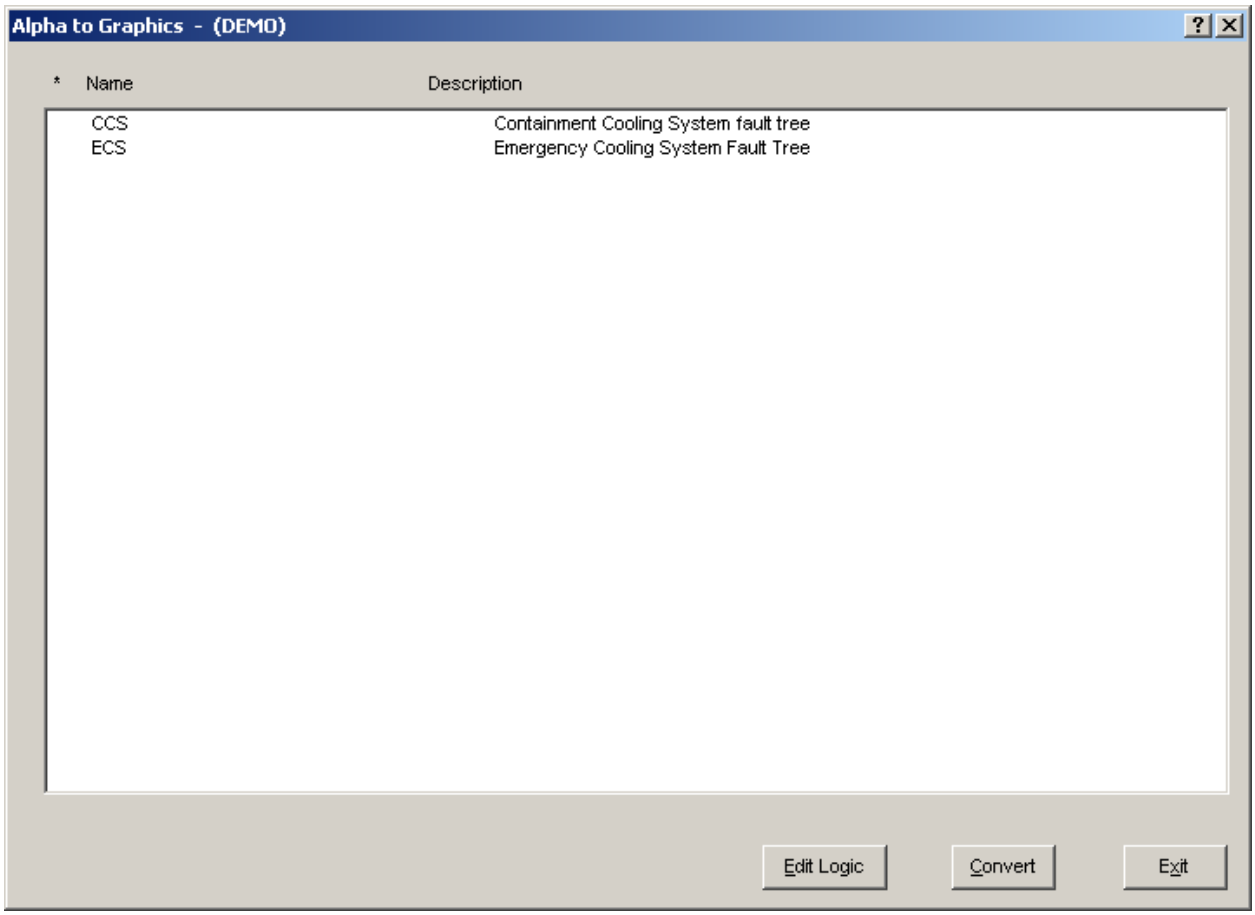
Using the Fault Tree Alpha-to-Graphics Feature

To use the Alpha to Graphics feature, select **Utility ➔ Fault Tree ➔ Alpha to Graphics** or **Utility ➔ A to G** button.





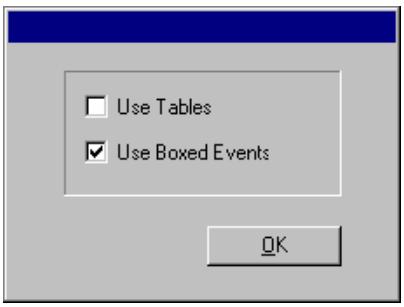
♦ Choose the **Alpha to Graphics** sub-menu option.



♦ Highlight the desired system(s) and choose the **Convert** button.

Specifying Alpha to Graphics Conversion Options

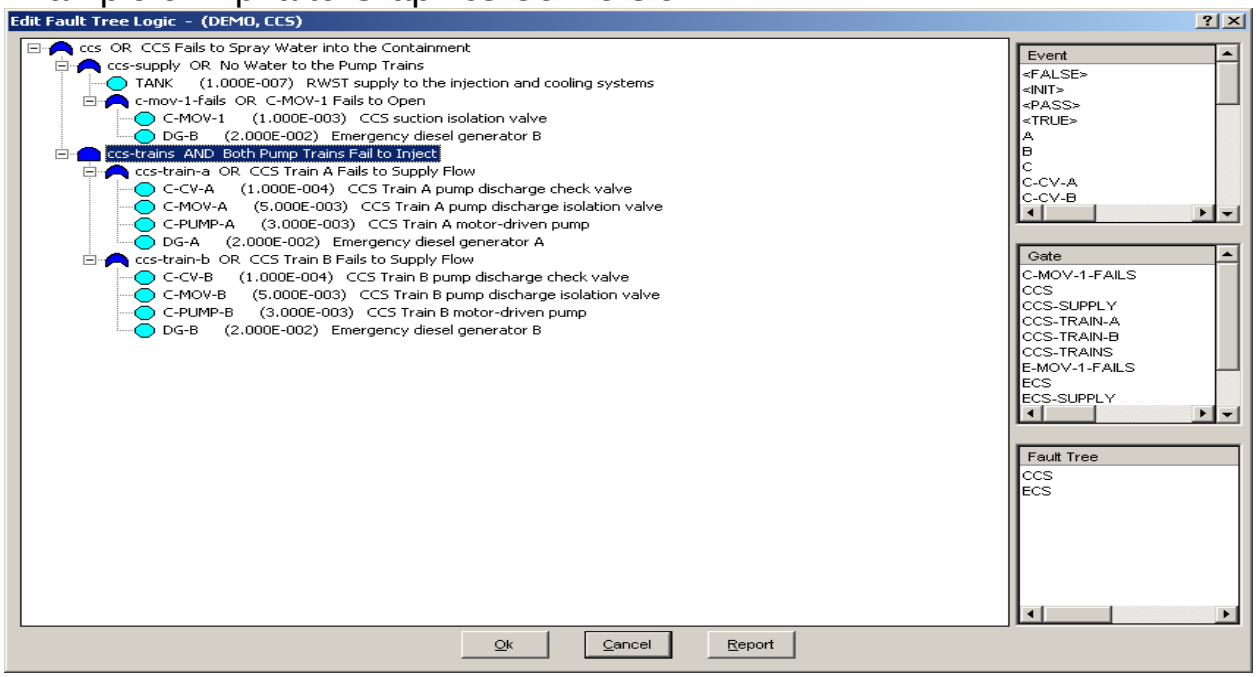
- To place basic events in the *Table of Basic Events* symbol, select the "Use Tables" check box.



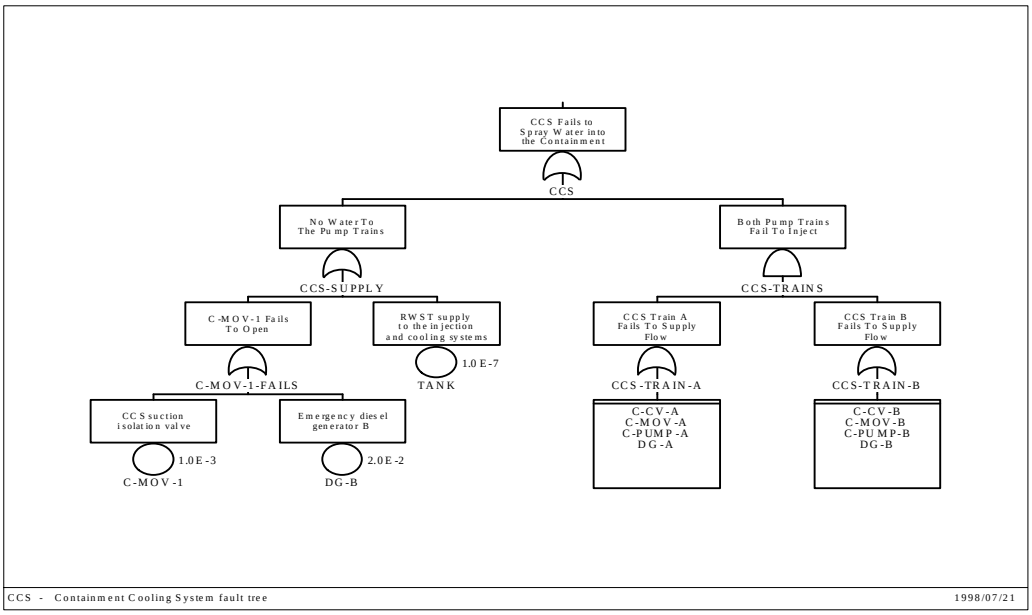
- You may also choose from the two basic event symbols, the round *Basic Event* symbol or the *Boxed Basic Event* symbol. To use the *Boxed Basic Event* symbol, select the "Use Boxed Events" check box.

Note: Special basic event symbols such as the Undeveloped Event symbol are not distinguished from ordinary basic events in the fault tree logic; therefore, the Alpha to Graphics conversion will *only* use the boxed, unboxed, or table basic event symbols.

Example of Alpha to Graphics Conversion



Alpha to Graphics Conversion



- After performing the Alpha to Graphics conversion, the fault tree logic is converted to a graphical fault tree. The conversion was performed using Tables and Boxed Basic Events.

  Gate and basic event descriptions will automatically be placed on the fault tree.

6.2. The Page Tree Feature

Function — The **Page Tree** option is used to permanently divide one fault tree "page" or file into several "pages" or files. This allows you to divide the contents of a fault tree file so that it fits onto a printed page. Dividing the fault tree is accomplished by picking gates to be changed into transfers.

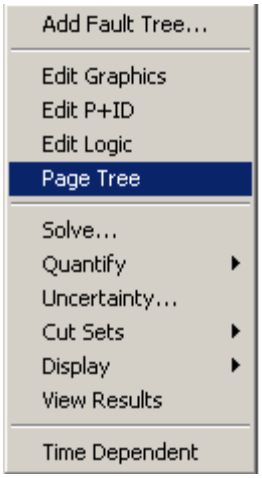
Additionally, you can break out a subtree at a user-specified gate, output gates with multiple references to a separate subtree, or output multiple top gates into separate fault trees.

When Using the Page Tree Feature

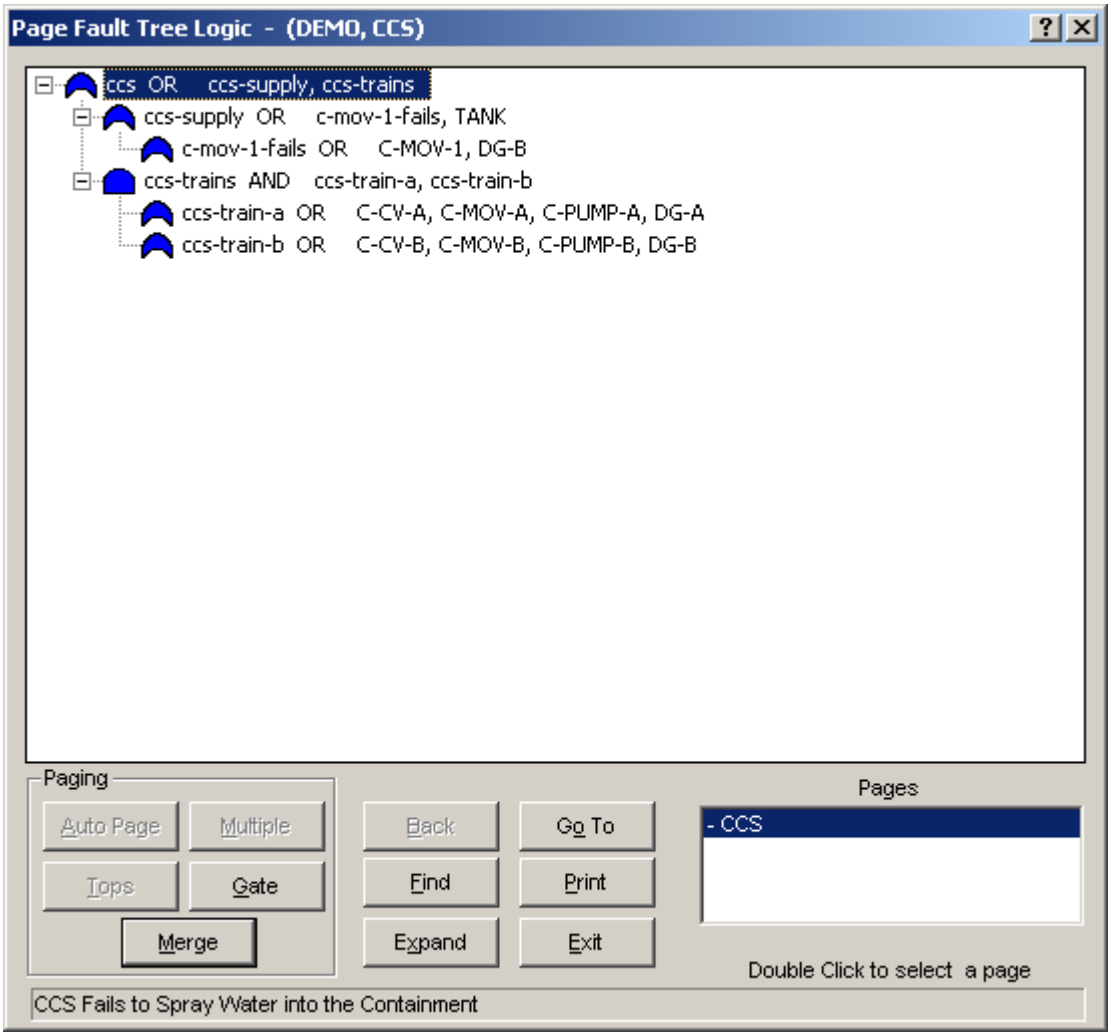
Always backup the database prior to using the Page Tree feature, because ***changes cannot be undone easily***. You are making changes to the fault tree logic just as if you went into the graphical editor and cut/paste new gates and transfers.

Using the Page Tree Feature

- To use the Page Tree feature, select the **Fault Tree** option from the menu bar.
- Highlight the fault tree to be paged and right-click to invoke the pop-up menu.
- Select the **Page Tree** option.



- The *Page Fault Tree Logic* dialog is displayed.



Page Options

Multiple - Break out gates with multiple references (i.e., gates that appear in more than one location within the fault tree) and output the logic into separate subtrees. Enabled only when multiply-referenced gates are detected.

Tops - Automatically separate the logic file into different fault trees by the top events. Enabled only when multiple tops in the logic are detected.

Gate - Create a separate fault tree logic file with the specified gate as the top event.

Find - Locate a specified gate.

Go To - Follow a transfer to another fault tree file.

Merge - Merge the logic represented by a transfer gate into the current fault tree file. This will occur only if the gate is not referenced more than once within the fault tree.

Auto Page - Automatically break the fault tree into easily viewable segments.

Expand - Expand all gates in the current fault tree file.

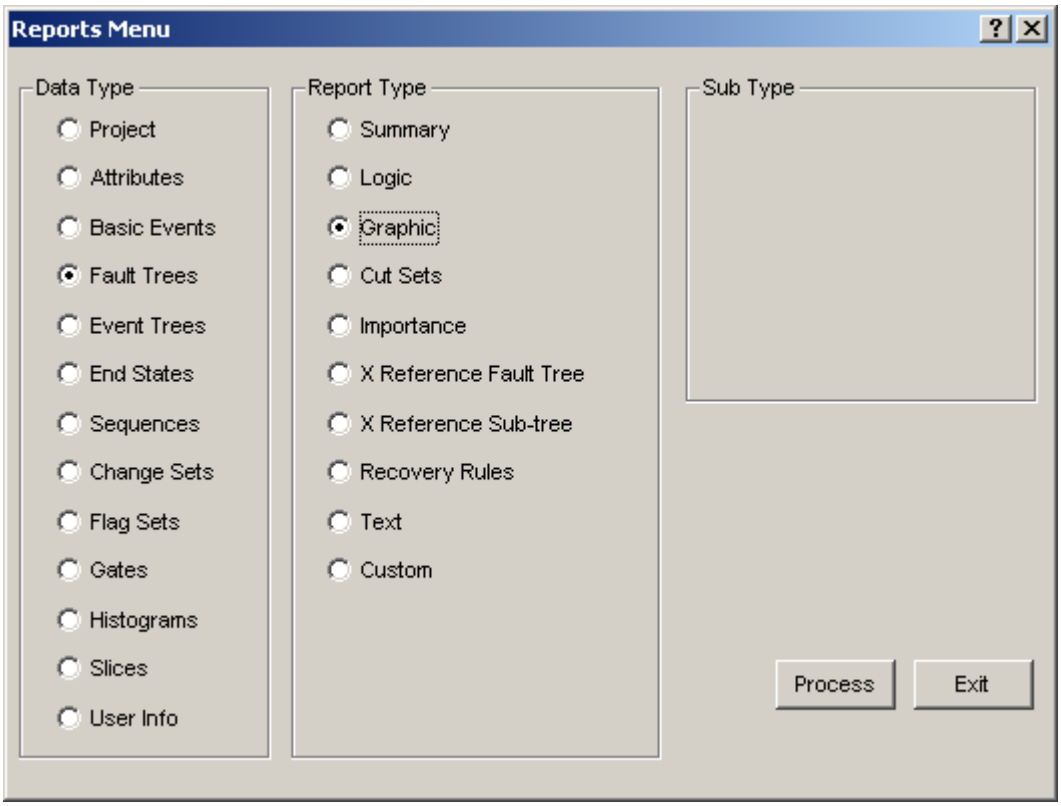
Back - Reload the previous fault tree file. This option is enabled only after executing the **Go To** option.

Pages - List of existing pages. Double-click on a page in the list to make it the current page.

Exit - Close the *Page System Logic* dialog. If changes were made to the system logic, you will be prompted to save the changes before exiting.

6.3. Printing Fault Tree Graphics

- To print fault tree graphics, select **Report ➔ Fault Tree ➔ Graphic**.



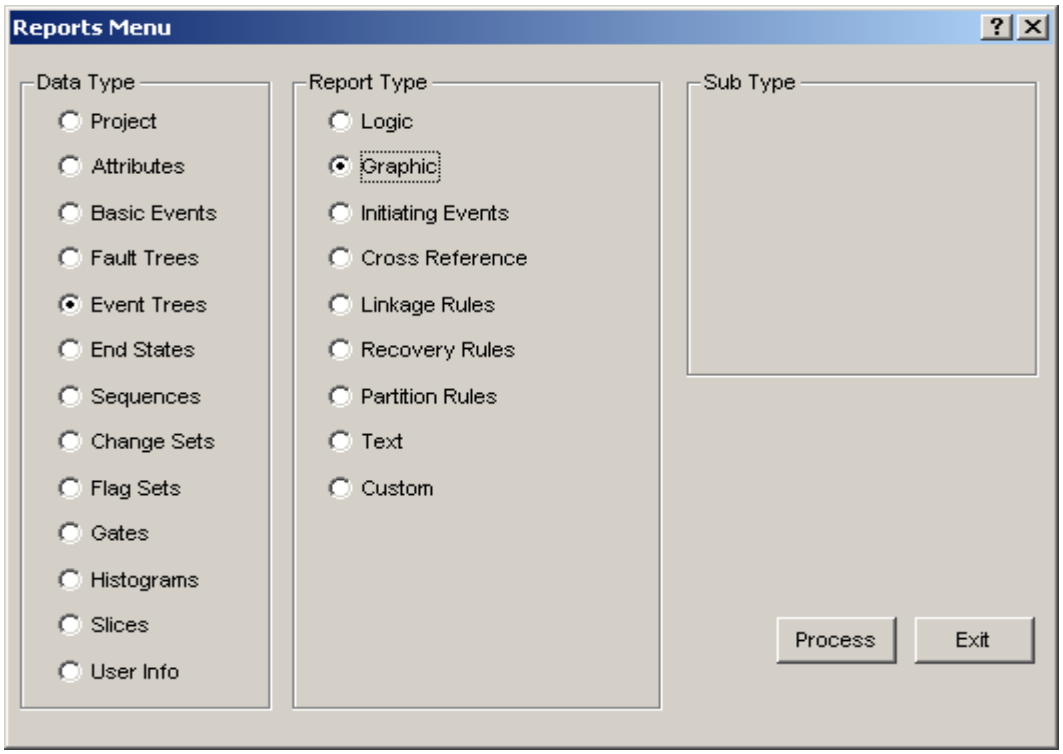
- Select the files to be prepared for plotting, and choose the **Print** button.

- Select the files to be prepared for plotting, and choose the **Export** button to create metafiles that can be loaded directly into a word processing program.

Note: the size and alignment of the graphic as it appears in the graphical editor prior to using the print option dictates the appearance of the printed graphic.

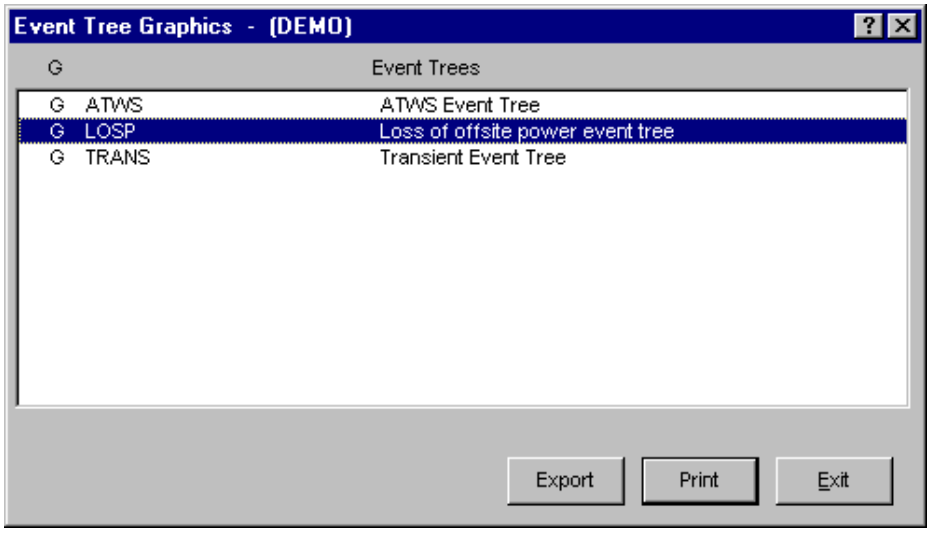
6.4. Printing Event Tree Graphics

♦ To print event tree graphics, select **Report ➔ Event Tree ➔ Graphic**.



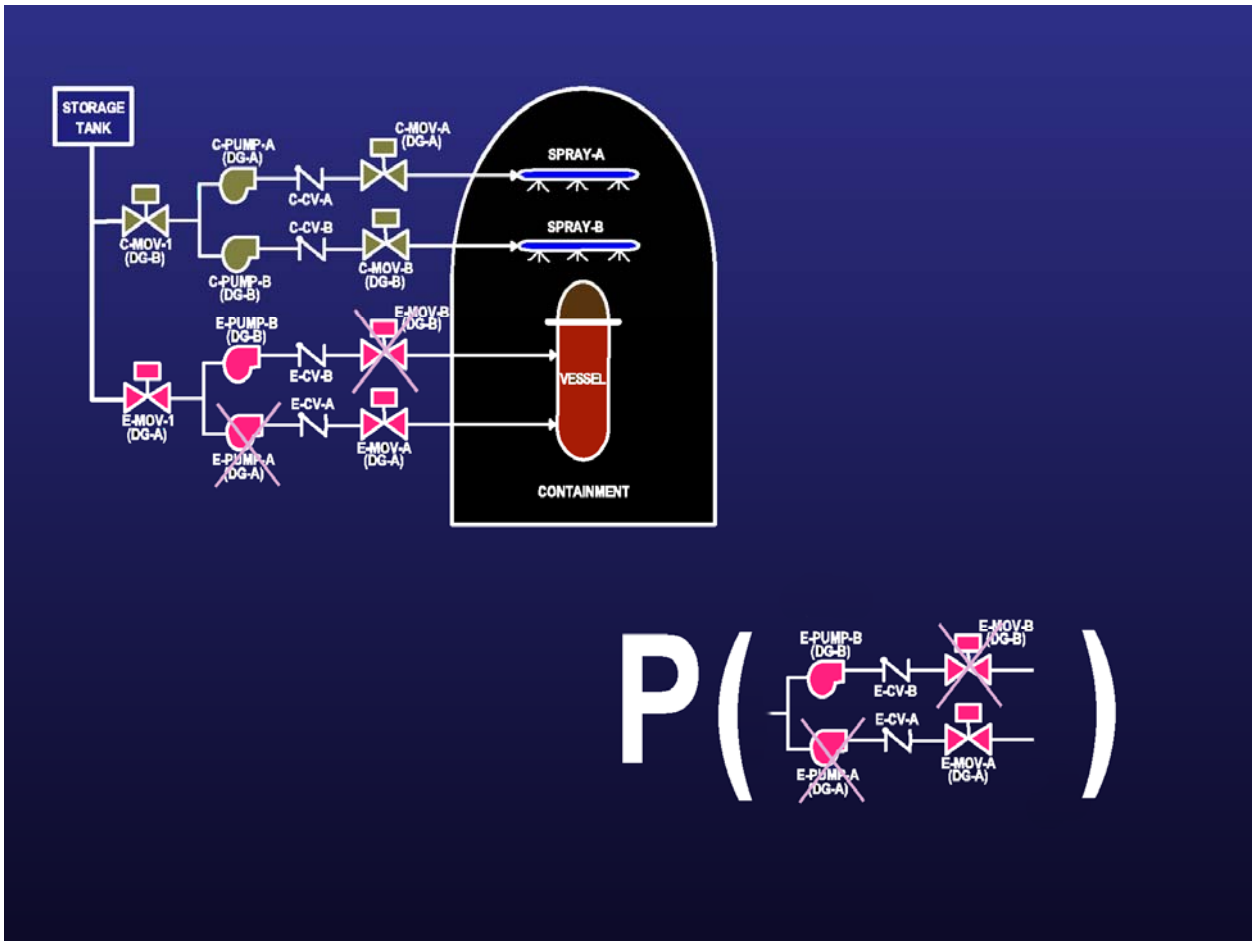
♦ Select the files to be prepared for plotting, and choose the **Print** button.

♦ Select the files to be prepared for plotting, and choose the **Export** button to create metafiles that can be loaded directly into a word processing program.



7 GENERATING FAULT TREE CUT SETS

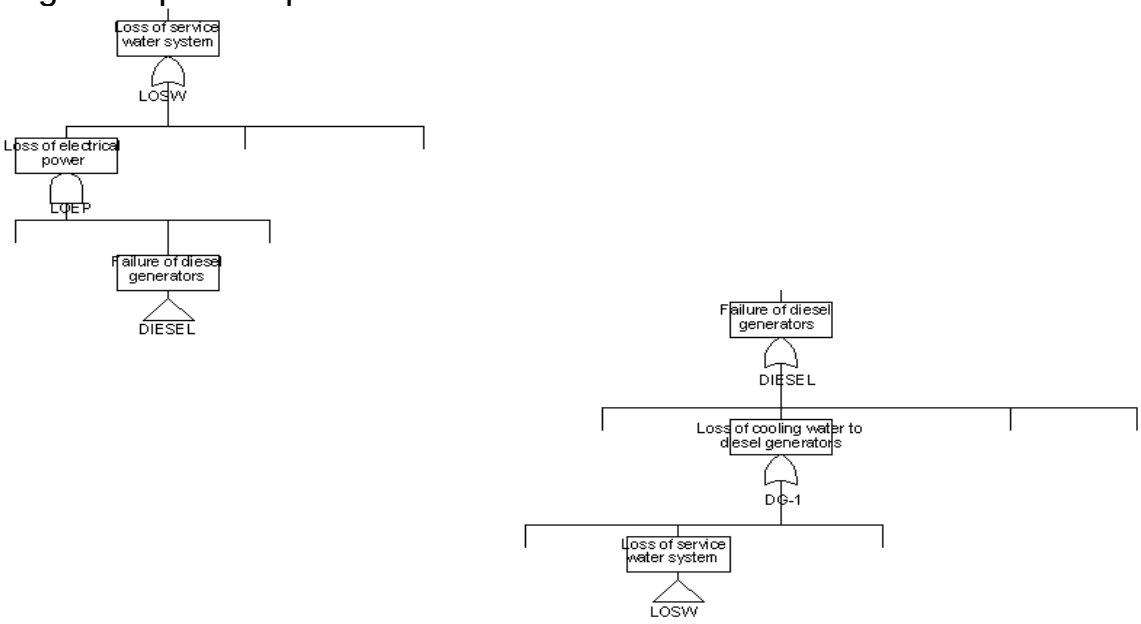
Section 7 describes how to generate fault tree cut sets. Model preparation prior to generating cut sets is discussed, and the various analysis and truncation options are described. Cut set display features are also presented.



7.1. Prerequisites for Generating Fault Tree Cut Sets

1. Fault tree logic was created by using the fault tree graphics editor, fault tree logic editor, or loaded into the database via the MAR-D interface.
2. Basic event data were added through the **Modify ➔ Basic Events** menu.
3. Basic event data was prepared for by using the **Generate** option.
4. Fault tree transfers are properly modeled so that there are no logic loops in the fault trees, there is only one top gate in each fault tree, and the naming of transfer gates and fault tree filenames is consistent.

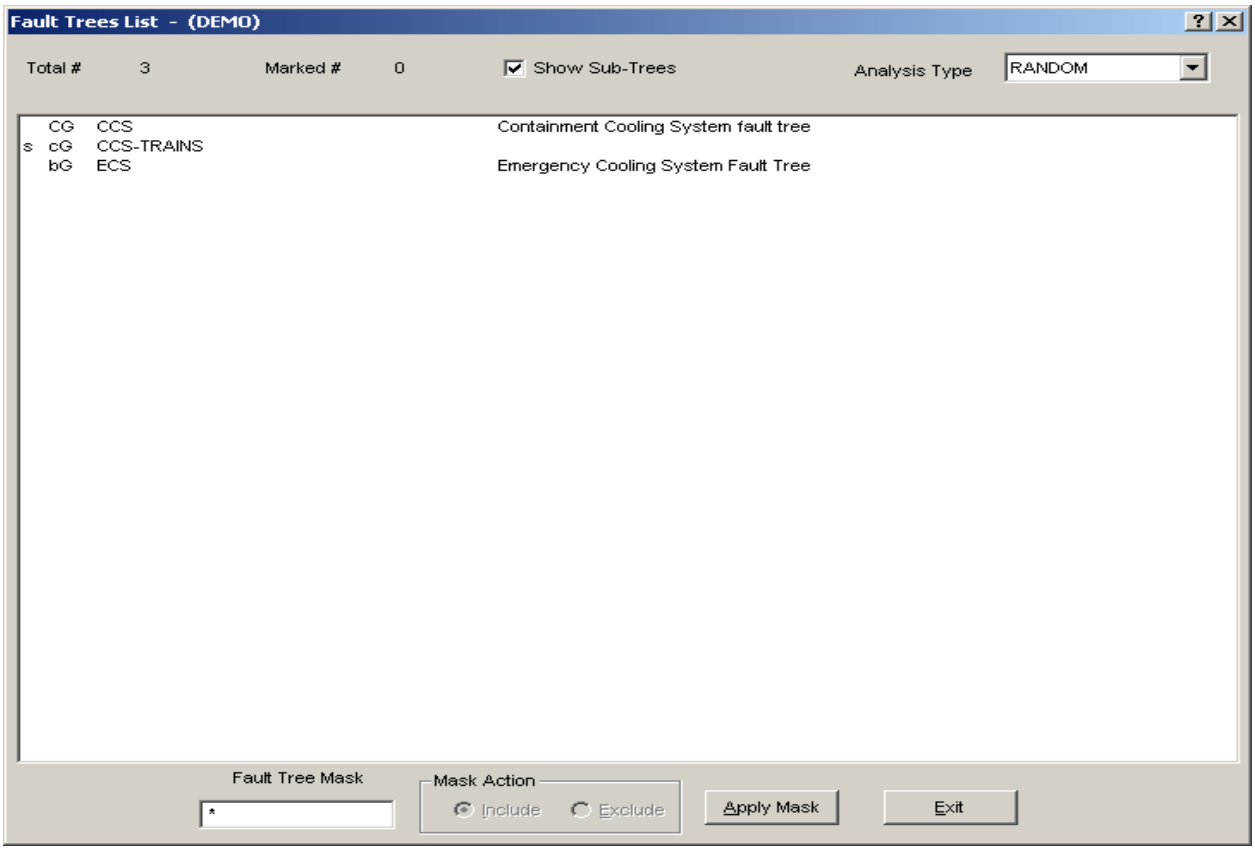
Logic Loop Example



The correct way to "break the loop" will depend on which system is being analyzed.

7.2. Menus and options for fault tree cut set generation

- Select the **Fault Tree** option from the menu.

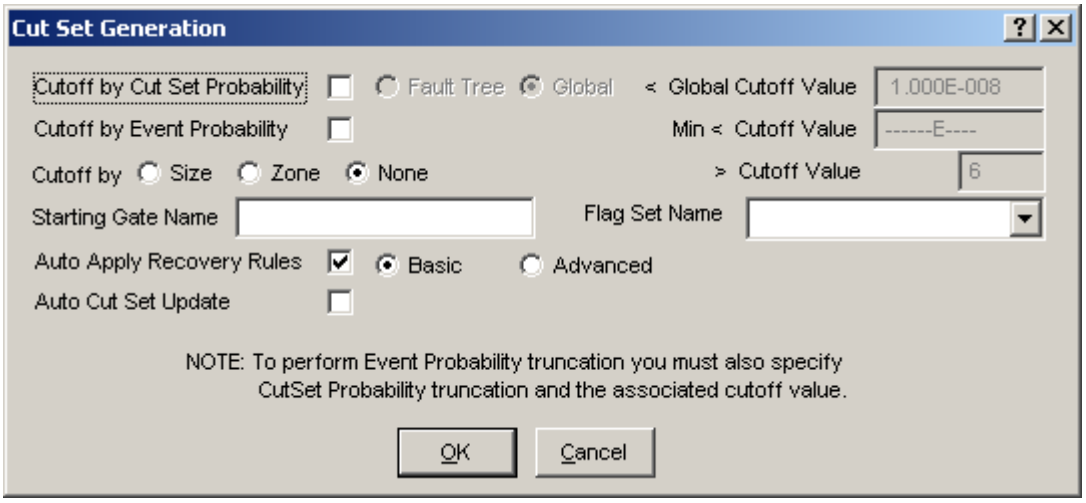


- Mark the Fault Trees using the mask feature, or using the mouse.
- Right-click to invoke the pop-up menu.
- Select the **Solve** option.

This option uses the fault tree logic from all fault trees that link to the top gate in the system. The fault tree probability is quantified using the minimal cut set upper bound.

b – flag indicates fault tree has base case cut sets
c – flag indicates fault tree has current case cut sets
C – flag indicates fault tree has both current and base case cut sets
s – flag indicates that the fault tree is a subtree (subtrees can be solved for cut sets)
G – flag indicates that graphics exist for the fault tree.

Truncation Parameters



Select the desired truncation parameters on the dialog, and choose **OK** to begin generating cut sets.

Cutoff by Cut Set Probability - If you select this check box, then cut sets below the cutoff value will not be retained. Choose one of the radio buttons:

> ***Global*** – uses the cutoff value in the ***"< Global Cutoff Value"*** field.
>
> ***Fault Tree*** – uses the cutoff value stored in the fault tree record (via the **Modify ➔ Fault Tree** option).

Cutoff by Event Probability - If you select this check box, then you must also select the ***Cutoff by Cut Set Probability*** check box. This option will retain cut sets comprised of basic events that are above the ***Min < Cutoff Value*** even if the cut set is below the ***Global Cutoff Value***. This option is generally not used.

Cut Set Size - If you select this check box, then cut sets having more events than specified in the ***> Cutoff Value*** field will not be retained. If you select the ***Zone*** check box, then cut sets having more Zone Flagged Events than specified in the ***> Cutoff Value*** field will not be retained. If neither check box is selected, then the number of events in a cut set will not affect whether the cut set is retained or discarded. This option is generally not used.

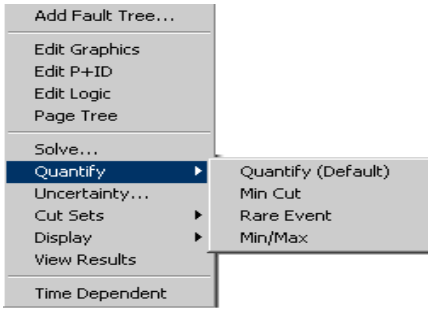
Starting Gate Name - If you leave the field blank, the top gate in the system will be used. If you specify a gate, that gate will be used as though it were the top gate. This option is generally not used.

Flag Set Name - If you leave the field blank, the system-specific flag set, if any, will be used. If you specify a flag set, that flag set will be used during processing. This option is generally not used.

Auto Apply Recovery Rules – If you check box, any recovery rules associated with this fault tree will automatically be applied after the fault tree cut sets have been generated.

Auto Cut Set Update – If this box is checked, then the existing current case cut sets are reevaluated to remove all non-minimal cut sets and the fault tree probability is requantified.

Quantify



This submenu provides options for requantifying existing current case fault tree cut sets. These options are designed to quickly requantify the cut sets when data changes have been made.

Minimal Cut Set Upper Bound Approximation

This calculation approximates the probability of the union of the minimal cut sets for the fault trees. The equation for the minimal cut set upper bound is

$$S = 1 - \prod_{i=1}^{m} (1 - C_i)$$

where

S = minimal cut set upper bound for the system unavailability,

C_i = probability of the I'th cut set, and

m = the number of cut sets.

Example: If the cut sets for a system are A, B, C, the system unavailability computed from the minimal cut set upper bound approximation is X = 1 - (1 - A)(1 - B)(1 - C).

Rare Event Approximation

This calculation simply sums each cut set as an approximation to the exact fault tree probability. This calculation is generally not used.

Min Max Quantification

The Min-Max quantification option quantifies the current case cut sets using the exact probability quantification algorithm. From the example above, the exact system unavailability is

$$X = (A + B + C) - (A * B + A * C + B * C) + (A * B * C),$$

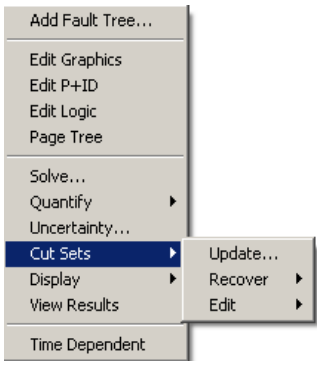
with the number of passes in this example being 3, corresponding to the number of pairs of parentheses.

Uncertainty

Performs Monte Carlo or Latin Hypercube uncertainty analysis for the selected fault tree. Fault tree uncertainty analysis is discussed further in Section 8.

Cut Sets

Performs manipulations on the fault tree cut sets.



Cut Set Update

This option uses the existing current case cut sets (unless the user specifies that base case cut sets are to be used instead). Non-minimal cut sets are eliminated and the fault tree probability is requantified.

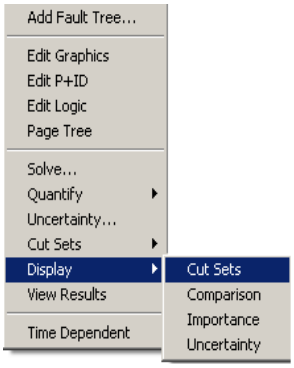
Recover Option

This option allows for post-processing rules to be created and applied to the fault tree cut sets. The rules can be directed at either a single fault tree or all fault trees.

Edit Option

This option provides a mechanism to manually edit the fault tree cut sets.

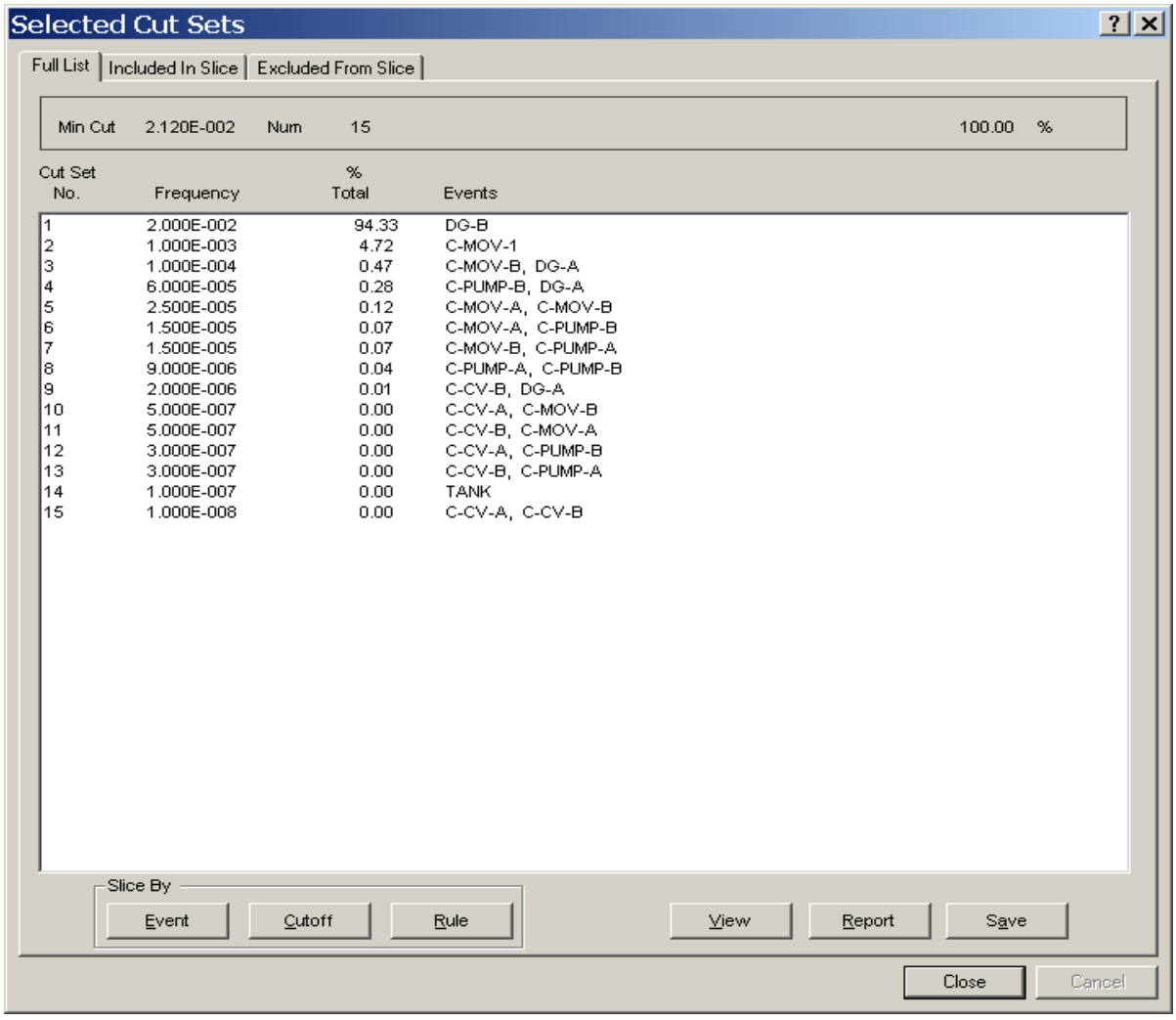
Displaying Fault Tree Cut Set Results



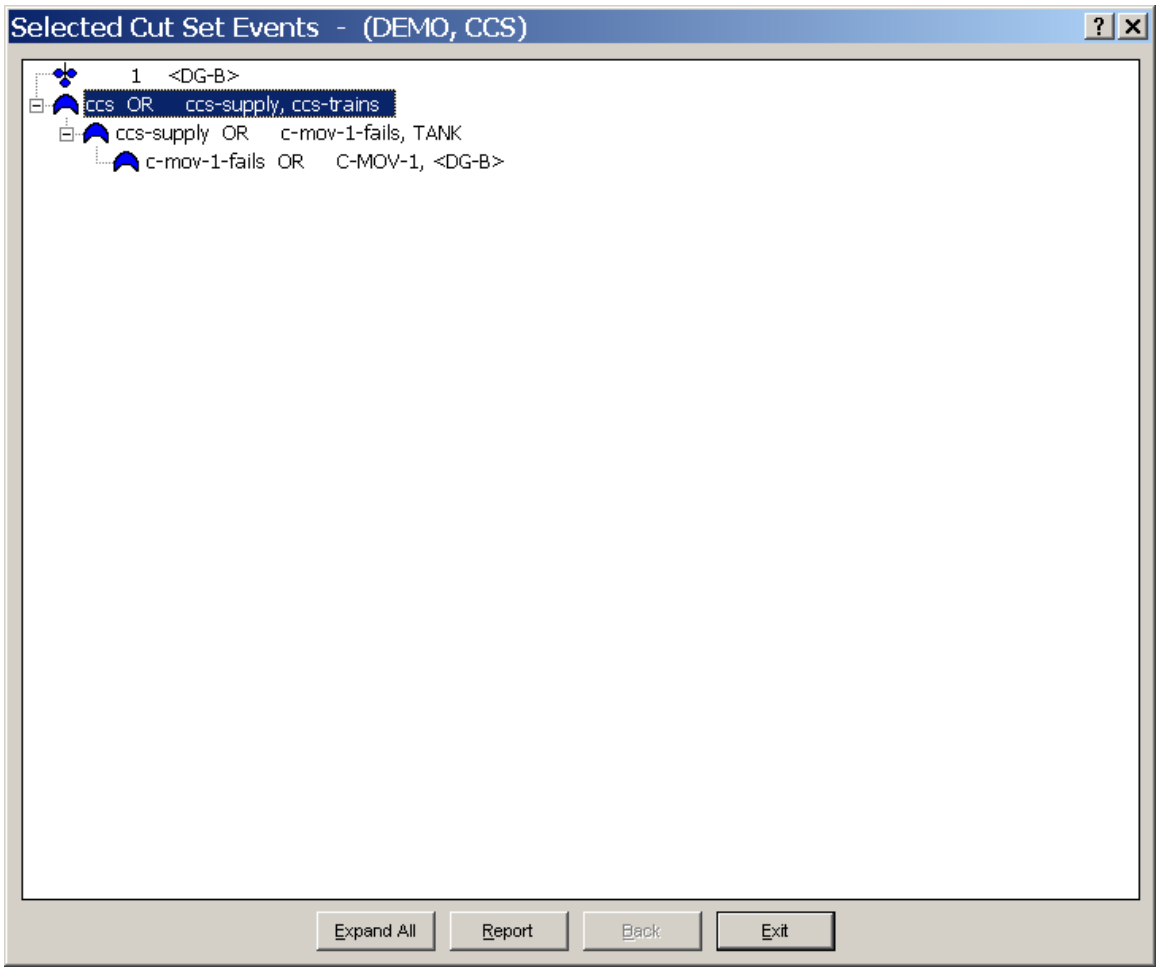
- To display fault tree cut sets, select **Fault Tree** from the menu bar.
- Highlight the fault tree that you want to view; right-click to invoke the pop-up menu.
- Select the **Display ➔ Cut Sets** option.

Selecting Fault Tree Cut Sets to View

- The fault tree cut sets and minimal cut set upper bound approximation of the fault tree failure probability are now displayed.
- To view the basic events in a cut set, highlight the cut set and click the **View** button.



- To find out where the events in the cut set came from, right-click on a cut set and select the **Path** option.



The Path option traces through the fault tree logic to indicate exactly where in the fault tree the cut set came from.

In the example above, the cut set is shown between brackets < >, DG-B comes from the CCS fault tree. CCS is an OR gate with inputs CCS-SUPPLY and CCS-TRAINS. From CCS-SUPPLY, we see a subgate called C-MOV-1-FAILS, which contains DG-B. Note that gates are lowercase while basic events are uppercase.

- The **Save** button from the cut set screen allows you to store the list of cut sets in a user-defined end state.

- The **Report** button from the cut set screen allows you to print a cut set report.

- The **Slice By** buttons from the cut set screen allows you to subdivide the list of cut sets into two lists based upon user-specified sort criteria. For example, if you wanted to show only cut sets that contained DG-A, enter the **Slice By Event** option and indicate that DG-A is the "selected event."

  The **Slice by Cutoff** button allows you to see the list of cut sets resorted by a different probability level (e.g., top X%, top 10 cut sets).

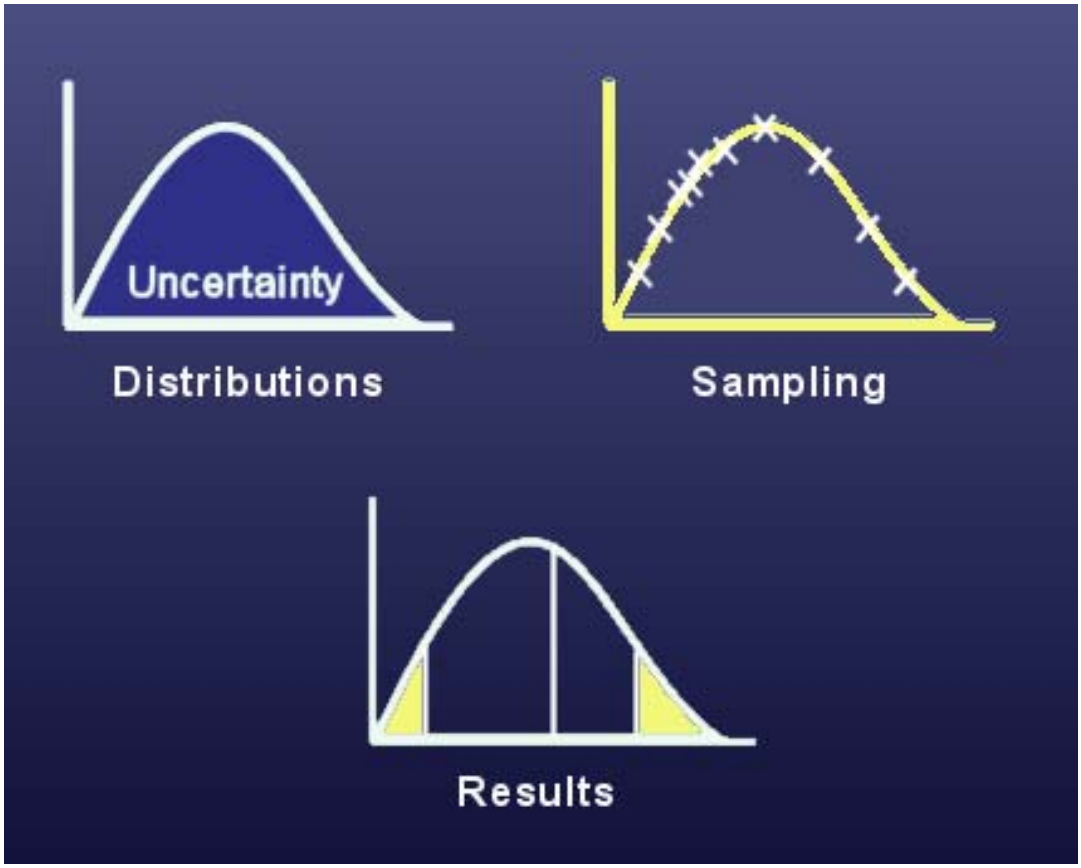
  The **Slice by Rule** button allows you to create a rule that will slice the cut sets up into specific groups.

  - If the **Slice** option is used, the **Included in Slice** tab lists the cut sets that were part of the slicing option and the **Excluded From Slice** tab lists the cut sets that were not part of the slice option.

Notes

8 FAULT TREE UNCERTAINTY ANALYSIS

Section 8 describes uncertainty analysis for fault trees. The concept of performing uncertainty analysis via Monte Carlo or Latin Hypercube sampling is discussed.



8.1. Fault Tree Uncertainty Analysis

Uncertainty analysis calculates the variability of a fault tree top event probability resulting from uncertainties in the basic event probabilities.

SAPHIRE provides two uncertainty analysis techniques:

Simple Monte Carlo sampling

Latin Hypercube sampling

Simple Monte Carlo Sampling

- A fundamental approach.
- Makes repeated quantifications of the system cut sets using each random variable sampled from the basic event uncertainty distributions.
- Requires more samples than Latin Hypercube sampling for the same degree of accuracy.

Latin Hypercube Sampling

- A stratified sampling technique, with the random variable distributions divided into equal probability intervals.
- Probability randomly selected from within each interval.
- May require fewer samples than simple Monte Carlo for similar accuracy; however, it may take longer to generate a random value than for a simple Monte Carlo sample.

8.2. Uncertainty Distributions for Basic Events

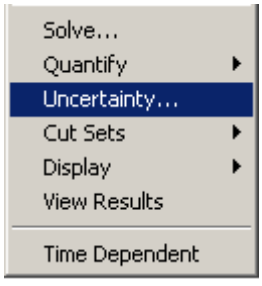
Basic event distribution types supported by SAPHIRE include:

Distribution	Code	Parameter
Blank		(Use point estimate value only)
Lognormal	L	Error factor
Normal	N	Standard deviation
Beta	B	B in Beta(a, b)
Dirichlet	D	Beta value of success branch
Gamma	G	r in $\Gamma(r)$
Chi-squared	C	Degrees of freedom
Exponential	E	(none)
Uniform	U	Upper end point
Histograms	H	Histogram number
Maximum Entropy	M	Lower and upper end point
Seismic	S	Beta r, Beta u
Constrained Noninformative	O	(none)
Triangular	T	Mode, Upper end point

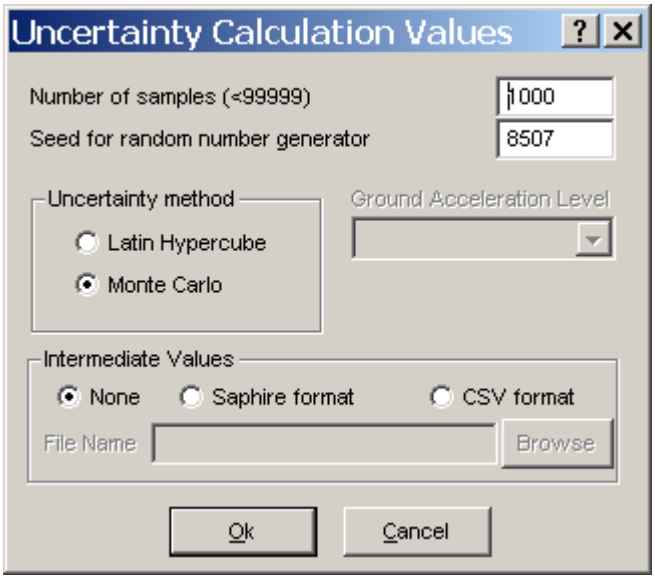
Correlation classes may be specified by the user to identify data (or other) dependencies for basic events using data derived from the same data source. Events in the same correlation class are 100% correlated.

8.3. Menus and Options for Performing Fault Tree Uncertainty Analysis

- Select **Fault Tree** from the menu bar.
- Mark the fault tree(s) for uncertainty analysis, right-click to invoke the pop-up menu.
- Select the **Uncertainty** option. The fault tree uncertainty will be calculated for each fault tree marked.



Uncertainty Calculation Values

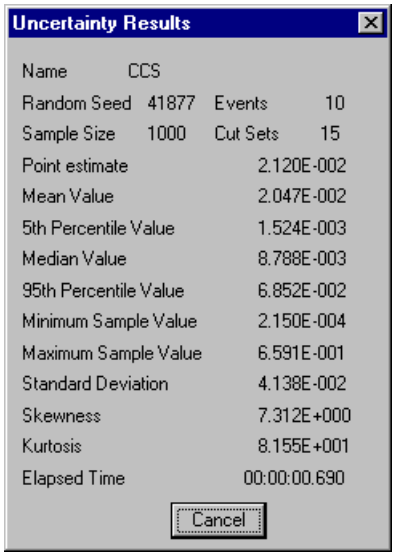


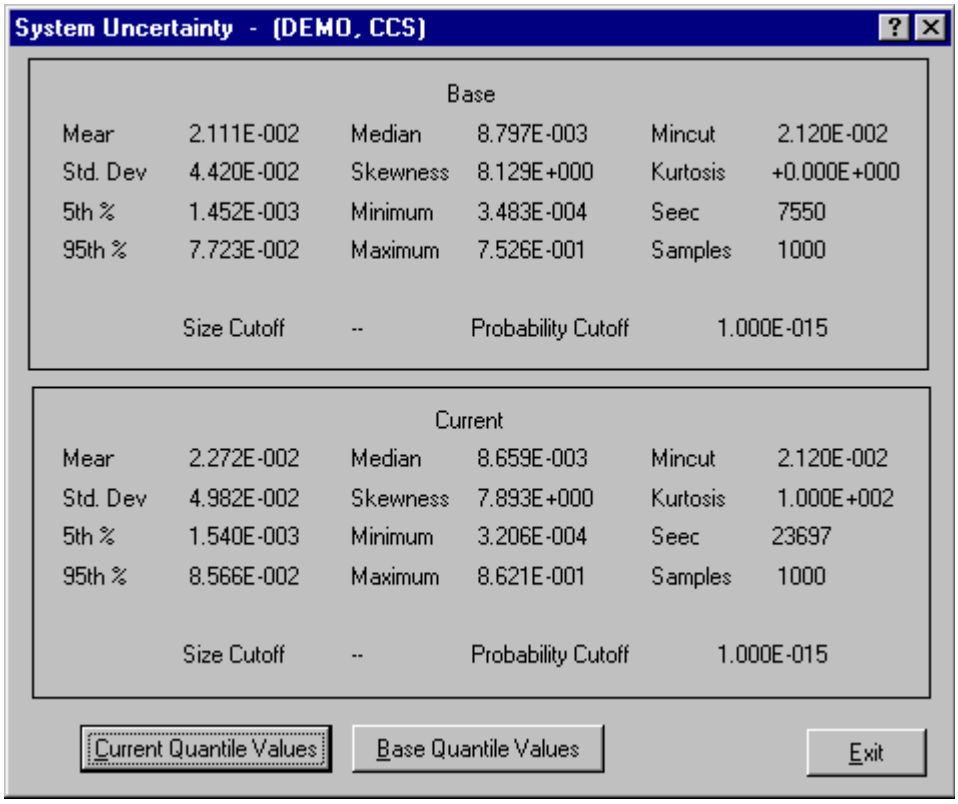
- Enter the uncertainty calculation options:
  - Select one of the uncertainty method radio buttons, Monte Carlo or Latin Hypercube.
  - Input the number of samples. (A larger number of samples will provide more accurate results but will require more computation time.)
  - Enter a value for the random number generator seed or accept the default. Enter zero to obtain a random seed from the system clock.
  - Enter a Ground Acceleration Level if seismic uncertainty is selected.

OPTIONAL: To save intermediate sample results to a disk file for review, select the Output Values check box and provide the Output file name

Uncertainty Results

- Uncertainty results will be displayed briefly on the screen following the uncertainty calculation.
- Uncertainty results for each fault tree can also be displayed from the **Fault Tree ➔ Display ➔ Uncertainty** menu.





Notes

9 FAULT TREE IMPORTANCE ANALYSIS

Section 9 describes the various fault tree importance measures available in SAPHIRE. Also, the calculations behind the importance measures are discussed.

9.1. Fault Tree Importance Measures

Importance measures provide “reliability-worth” information about basic events appearing in the cut sets for a fault tree.

Components showing high relative importance may be candidates for either (1) close monitoring to ensure that the component does not degrade over time or (2) design changes to increase the component reliability.

Ratio, Interval and Uncertainty importances are calculated for the highlighted fault tree cut sets.

- *Ratio Importance* — Fussell-Vesely Importance, Risk Reduction Ratio, and Risk Increase Ratio.

- *Interval Importance* — Birnbaum Importance, Risk Reduction Interval, Risk Increase Interval.

- *Uncertainty Importance* – This importance measure provides information about the uncertainty of the component (i.e., those components that will contribute the largest uncertainty to an uncertainty analysis of the selected fault tree cut sets).

9.2. Definitions of the Importance Measures

Fussell-Vesely Importance (FV)

An indication of the fractional contribution of the basic event to the minimal cut set upper bound. The equation for FV importance (of the i'th basic event) is

$$FV_i = F_i(x)/F(x)$$

where F(x) is the original minimal cut set upper bound
$F_i(x)$ is the minimal cut set upper bound with only the basic event of interest.

Risk Reduction Ratio (RRR) or Risk Reduction Interval (RRI)

An indication of how much the minimal cut set upper bound would decrease if the basic event probability were reduced (to a probability of 0.0. [i.e. never failed])

$$RRR = F(x)/F(0)$$

$$RRI = F(x) - F(0)$$

(Note the similarity between RRI and FV; the relative importance ranking of basic events will be the same for the two importance measures.)

Risk Increase Ratio (RIR) or Risk Increase Interval (RII)

An indication of how much the minimal cut set upper bound would increase if the basic event probability were increased (to a probability of 1.0).

$$RIR = F(1)/F(x)$$

$$RII = F(1) - F(x)$$

where F(x) is the original minimal cut set upper bound
F(1) is the minimal cut set upper bound with the event probability set equal to 1.0.

Birnbaum Importance (B)

Indicates the sensitivity of the minimal cut set upper bound with respect to a change in the basic event probability.

$$B = F(1) - F(0)$$

where F(1) is the minimal cut set upper bound with the event probability set equal to 1.0.

F(0) is the minimal cut set upper bound with the event probability set equal to 0.0.

Uncertainty Importance

The uncertainty in each input parameter, as expressed through its probability distribution, contributes to the uncertainty in the output parameter of interest (e.g., core damage frequency, loss of mission). The uncertainty importance measure in SAPHIRE quantifies the contribution of each individual basic event's uncertainty to this total output uncertainty.

Some useful importance measures relationships:

Birnbaum importance is equal to the sum of the RII and RRI importances.

$$B = RII + RRI$$

Fussell-Vesely (FV) importance is equal to the product of the Birnbaum importance and the event probability, divided by the minimal cut set upper bound.

$$FV = (B * x)/F(x)$$

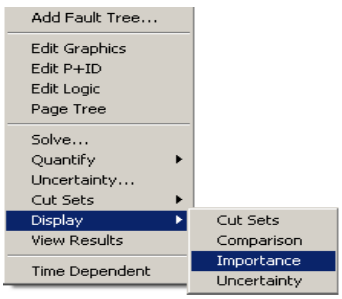
RRI importance is equal to the product of the Birnbaum importance and the nominal basic event probability (RRI importance is sometimes referred to as the inspection importance).

$$RRI = B * x$$

RII importance is equal to the product of the Birnbaum importance and the complement of the basic event probability.

$$RII = B * (1 - x)$$

9.3. Importance Menus and Options

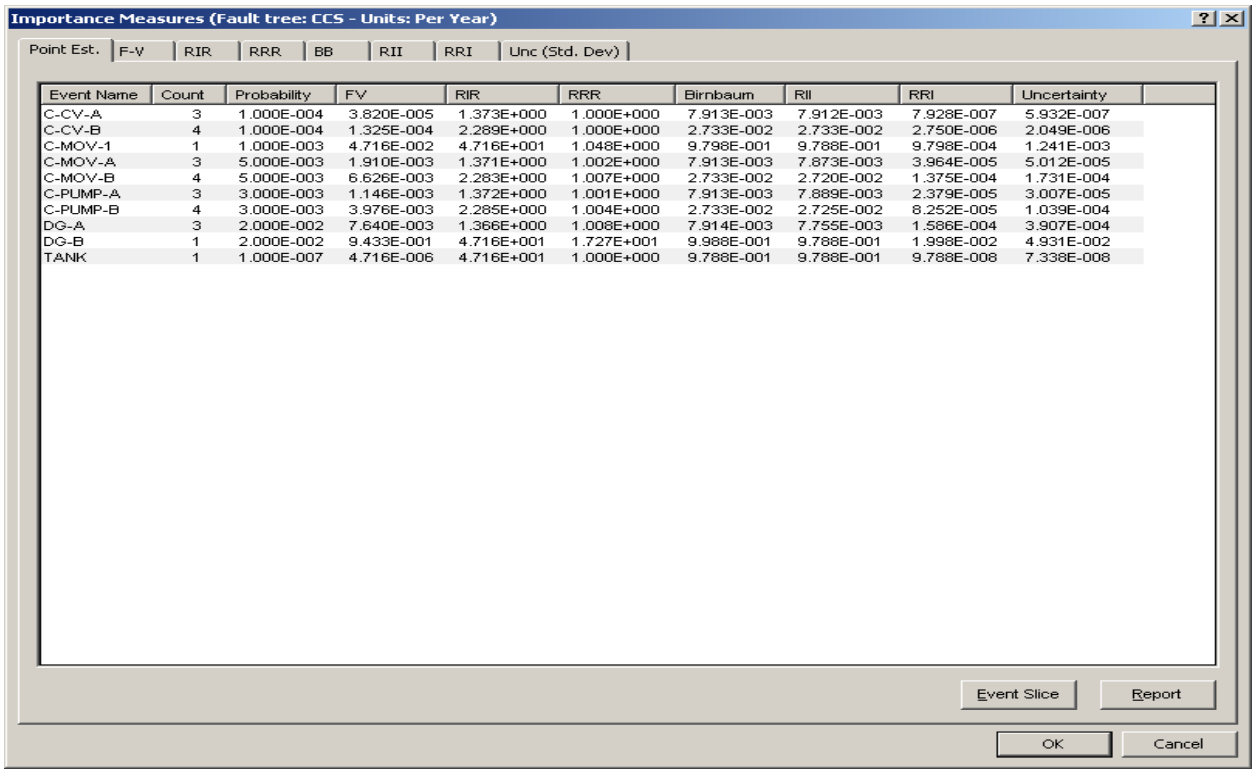


♦ Select **Fault Tree** from the menu bar.

- Highlight the fault tree, right-click to invoke the pop-up menu.
- Select **Importance.**

Viewing Importance Measures

- The importance measures are now displayed.



Event Name	Count	Probability	FV	RIR	RRR	Birnbaum	RII	RRI	Uncertainty
C-CV-A	3	1.000E-004	3.820E-005	1.373E+000	1.000E+000	7.913E-003	7.912E-003	7.928E-007	5.932E-007
C-CV-B	4	1.000E-004	1.325E-004	2.289E+000	1.000E+000	2.733E-002	2.733E-002	2.750E-006	2.049E-006
C-MOV-1	1	1.000E-003	4.716E-002	4.716E+001	1.048E+000	9.798E-001	9.788E-001	9.798E-004	1.241E-003
C-MOV-A	3	5.000E-003	1.910E-003	1.371E+000	1.002E+000	7.913E-003	7.873E-003	3.964E-005	5.012E-005
C-MOV-B	4	5.000E-003	6.626E-003	2.283E+000	1.007E+000	2.733E-002	2.720E-002	1.375E-004	1.731E-004
C-PUMP-A	3	3.000E-003	1.146E-003	1.372E+000	1.001E+000	7.913E-003	7.889E-003	2.379E-005	3.007E-005
C-PUMP-B	4	3.000E-003	3.976E-003	2.285E+000	1.004E+000	2.733E-002	2.725E-002	8.252E-005	1.039E-004
DG-A	3	2.000E-002	7.640E-003	1.366E+000	1.008E+000	7.914E-003	7.755E-003	1.586E-004	3.907E-004
DG-B	1	2.000E-002	9.433E-001	4.716E+001	1.727E+001	9.988E-001	9.788E-001	1.998E-002	4.931E-002
TANK	1	1.000E-007	4.716E-006	4.716E+001	1.000E+000	9.788E-001	9.788E-001	9.788E-008	7.338E-008

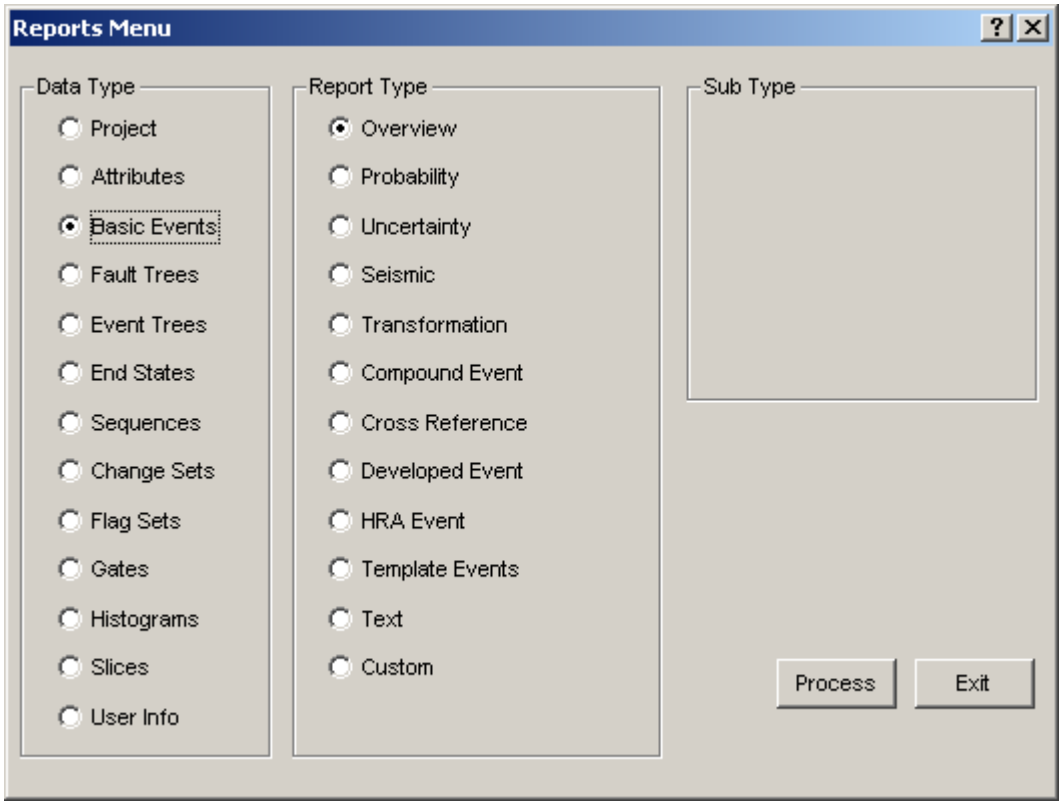
- Use the Report button on this dialog to write the importance measure results to a file or printer.
  - ◊ Alternately, the **Report ➔ Fault Tree ➔ Importance** option can be used to report fault tree importances.
- To sort by a column heading, click on the desired column heading. SAPHIRE then sorts the results for the selected importance measure in a descending order.
- To return to the Fault Tree List, click **OK** or **Cancel**.

10 REPORTING INFORMATION

Section 10 describes how to generate reports for a variety of information including basic event data and fault tree results stored in SAPHIRE.

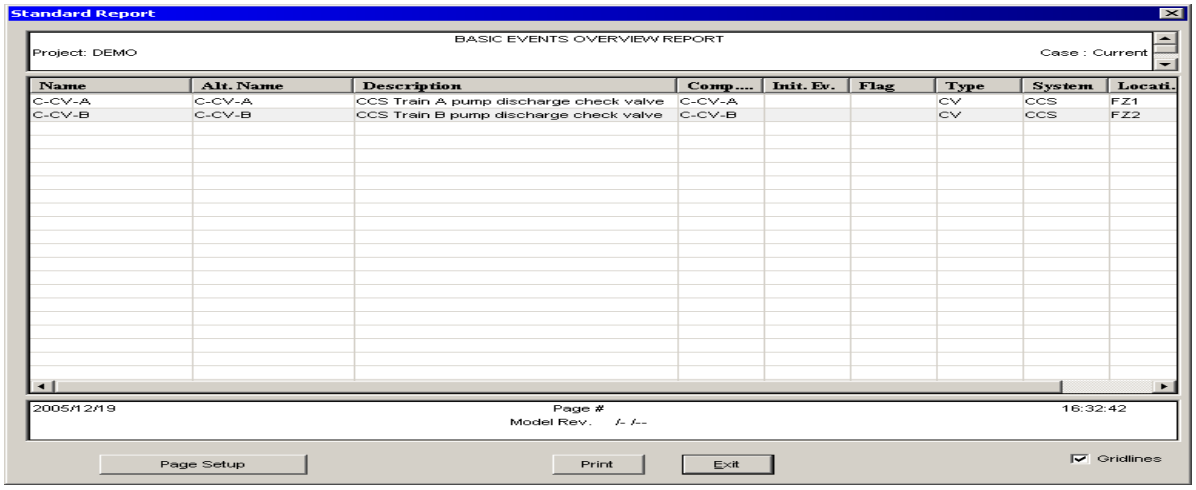
10.1. Basic Event Report Information

- To report basic event information, select **Report** from the menu bar.

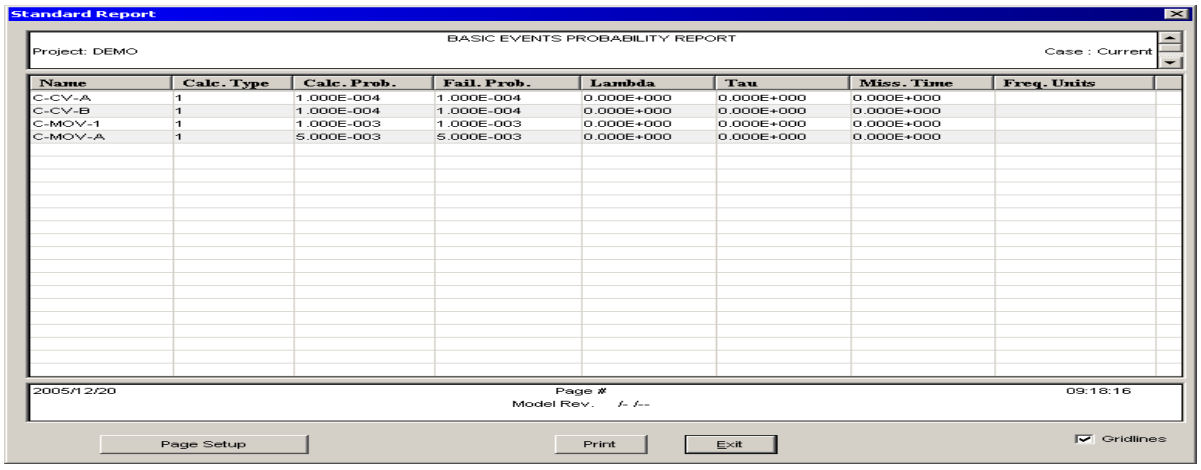


- Select **Basic Events** and then you can choose from several different report options.

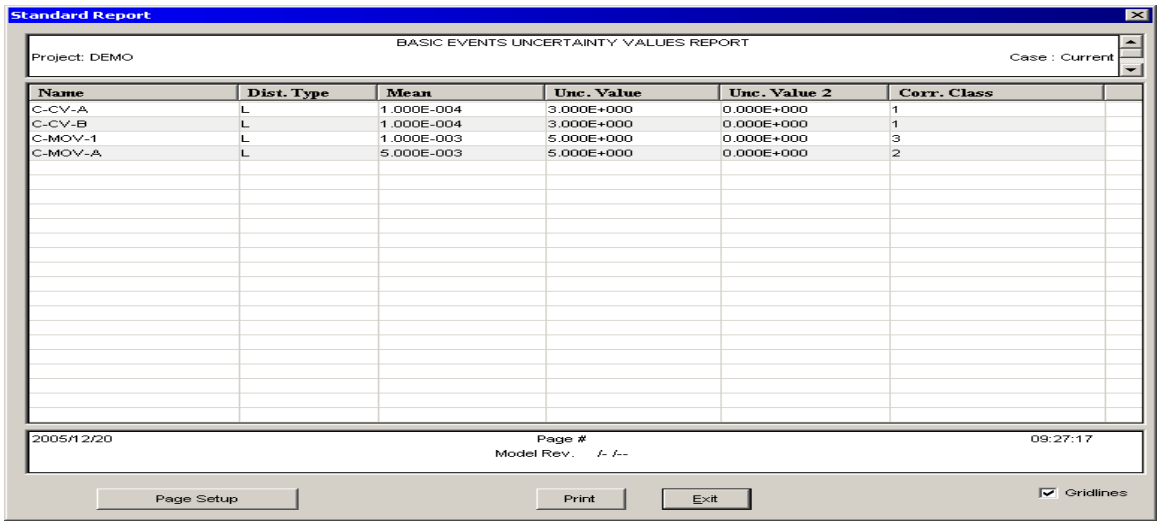
Basic Events Overview Report



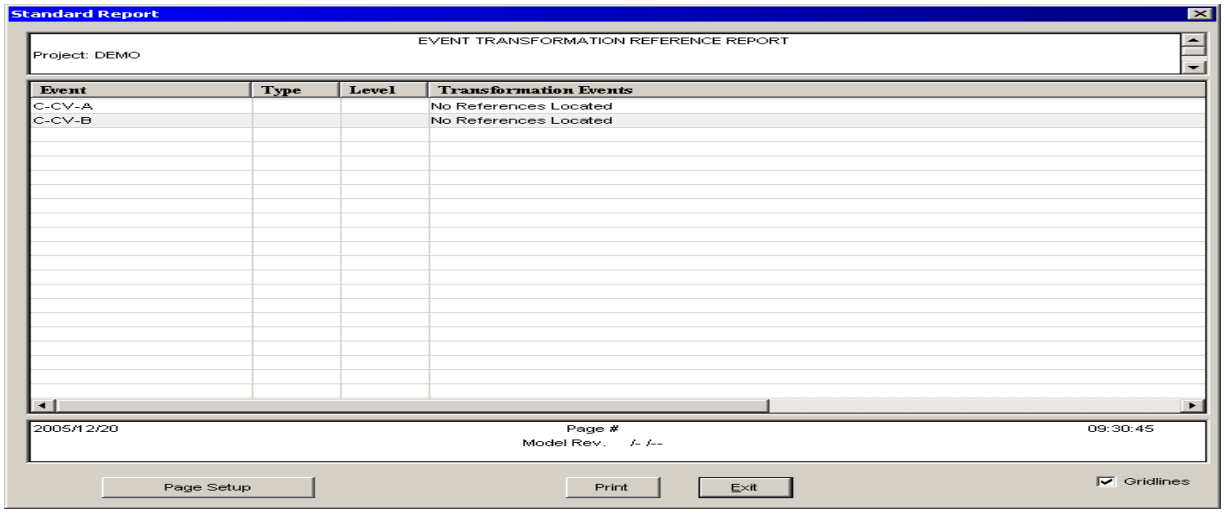
Basic Events Probability Report



Basic Events Uncertainty Values Report

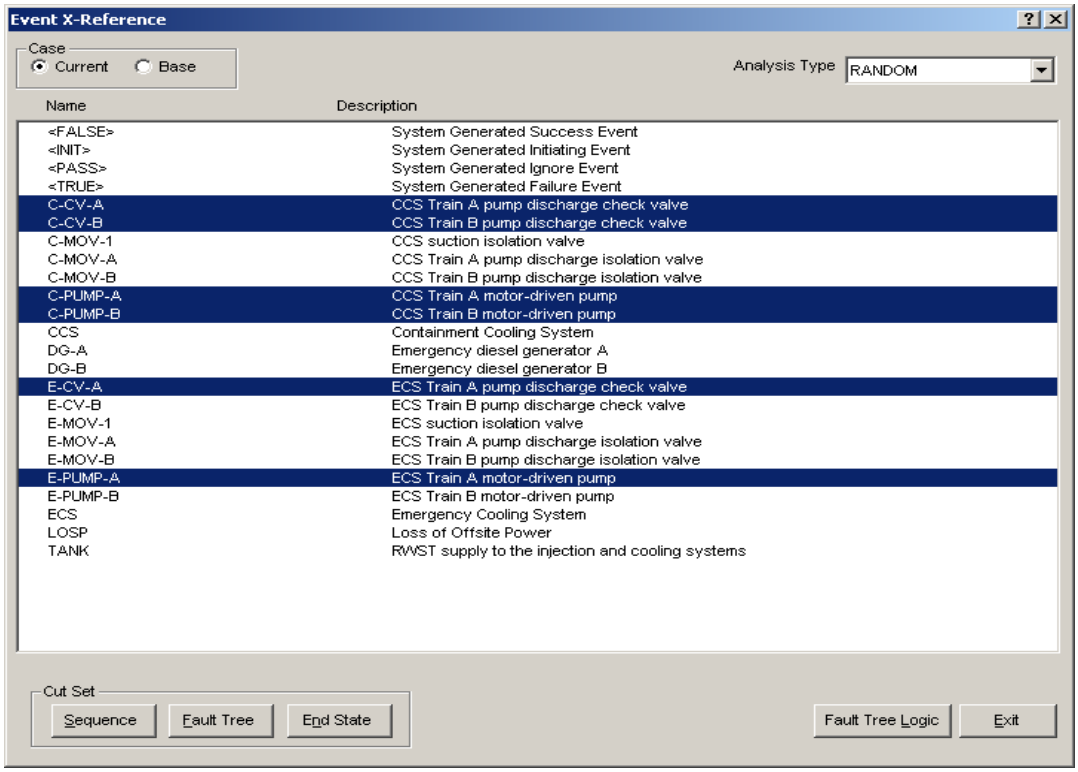


Event Transformation Reference Report

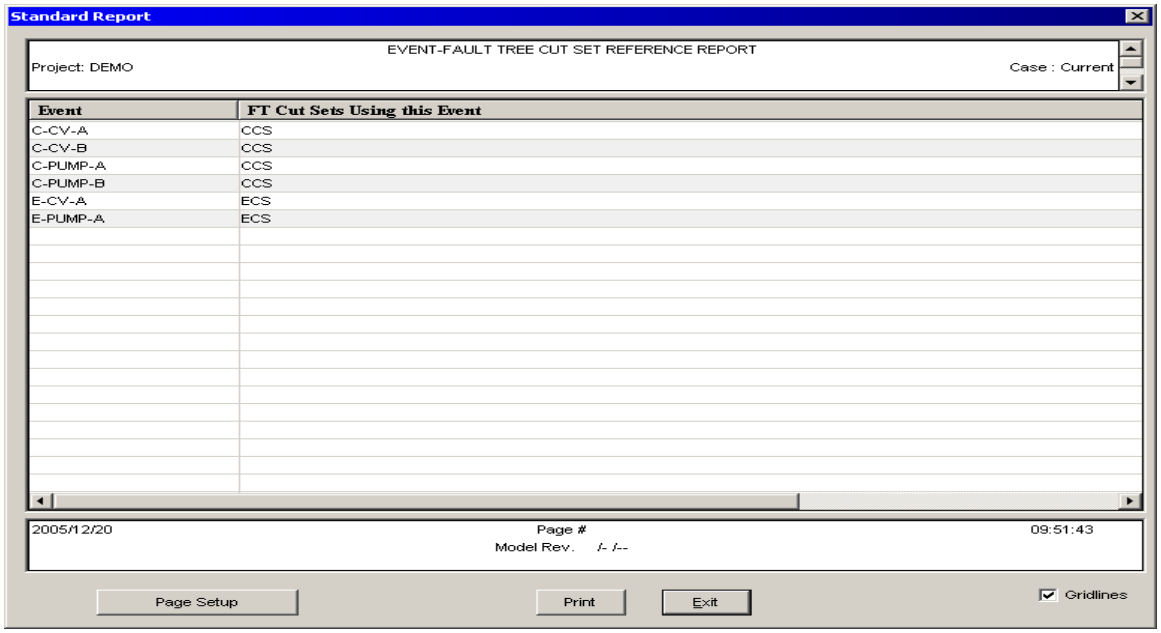


Event Cross-Reference Report

Several Cross Reference reports are available. Choose from sequence, fault tree, or end state cut set cross reference reports, or the fault tree logic cross reference report.

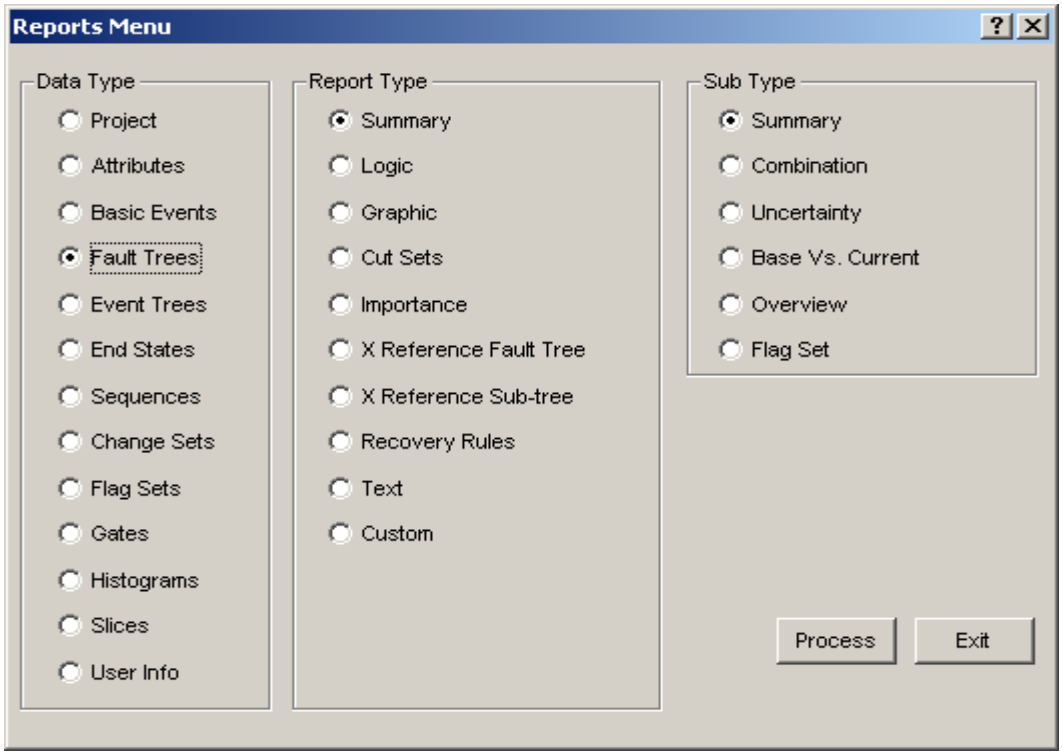


Event-Fault Tree Logic Reference Report



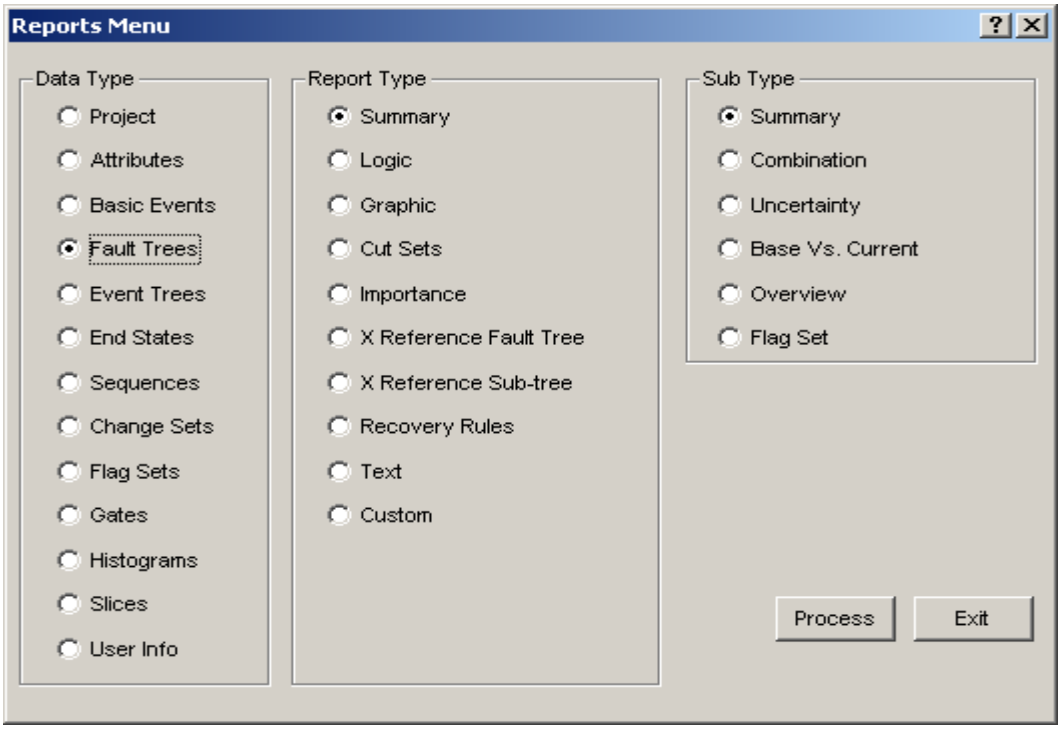
10.2. REPORTING FAULT TREE RESULTS

Fault Tree Report Menus and Examples

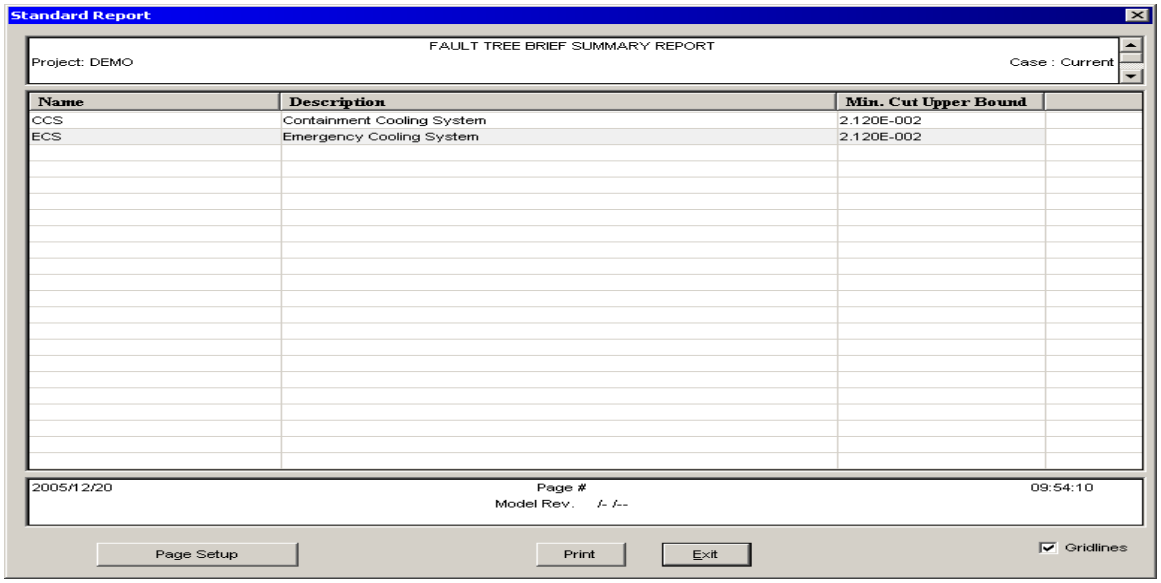


- To report fault tree information, select **REPORTS** from the main menu.
- Select **Fault Tree** and then you can choose from several different report options.

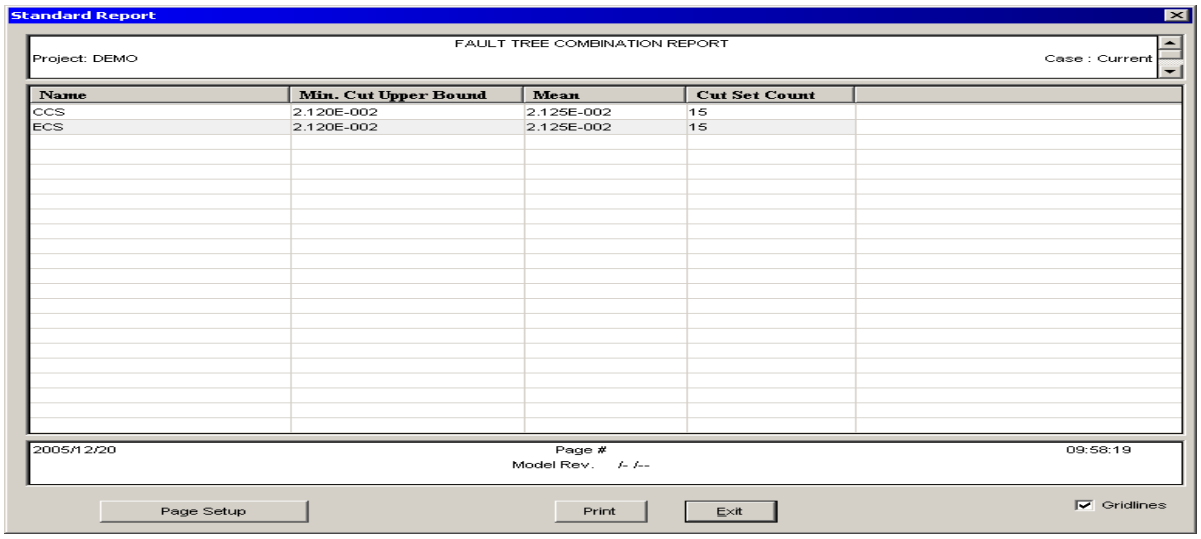
Fault Tree Summary Reports



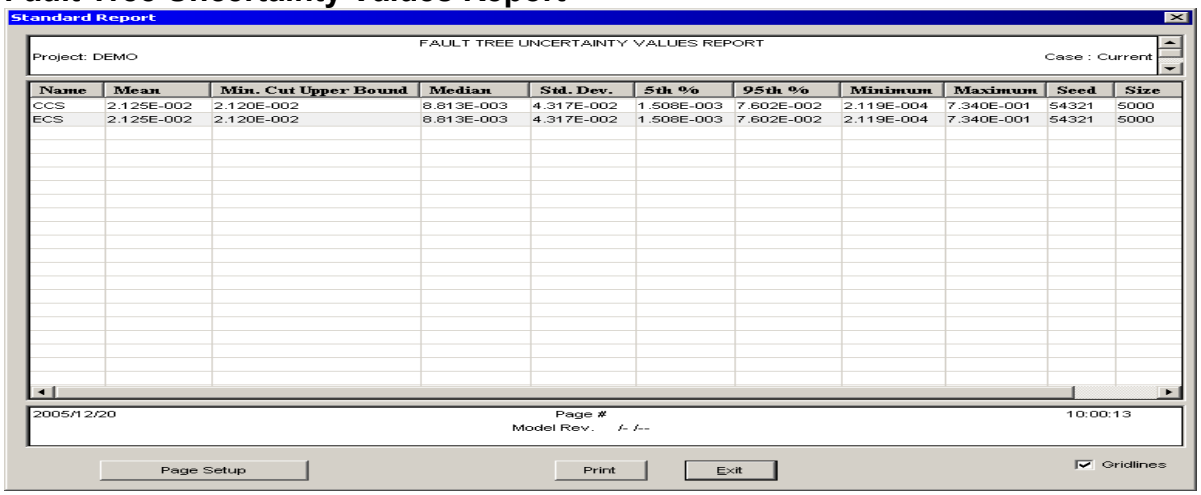
Fault Tree Brief Summary Report



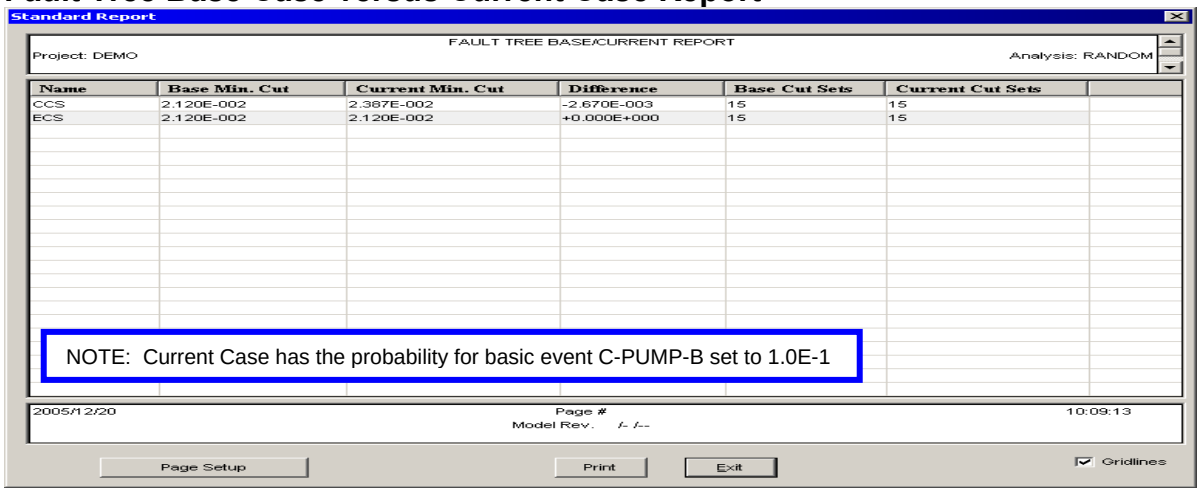
Fault Tree Combination Report



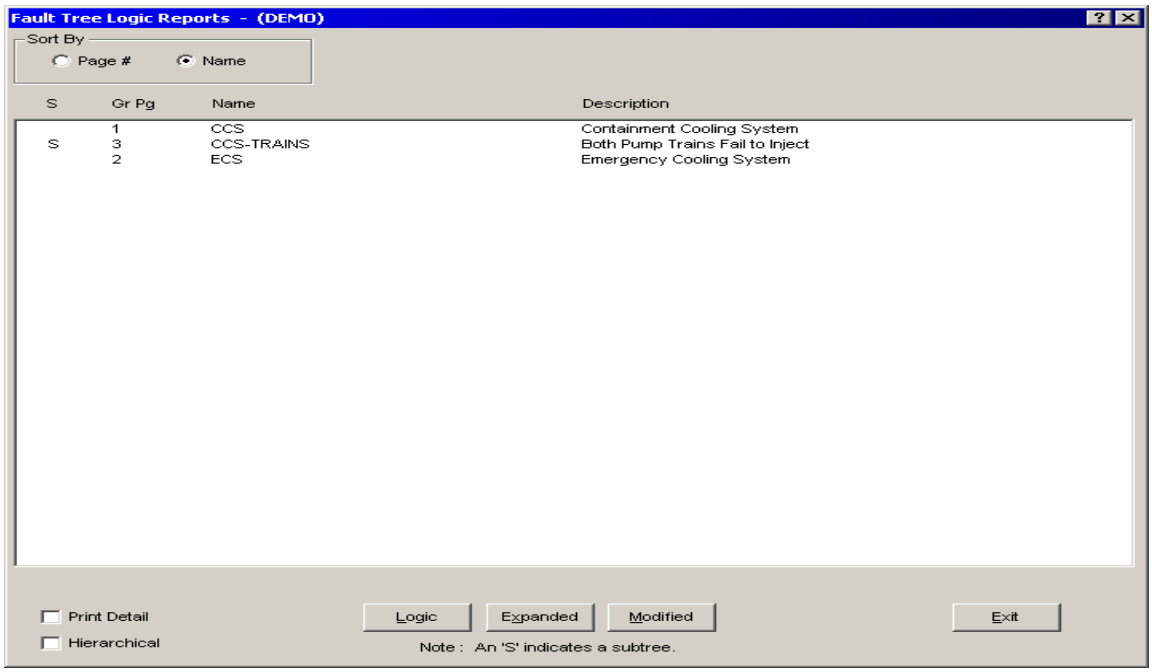
Fault Tree Uncertainty Values Report



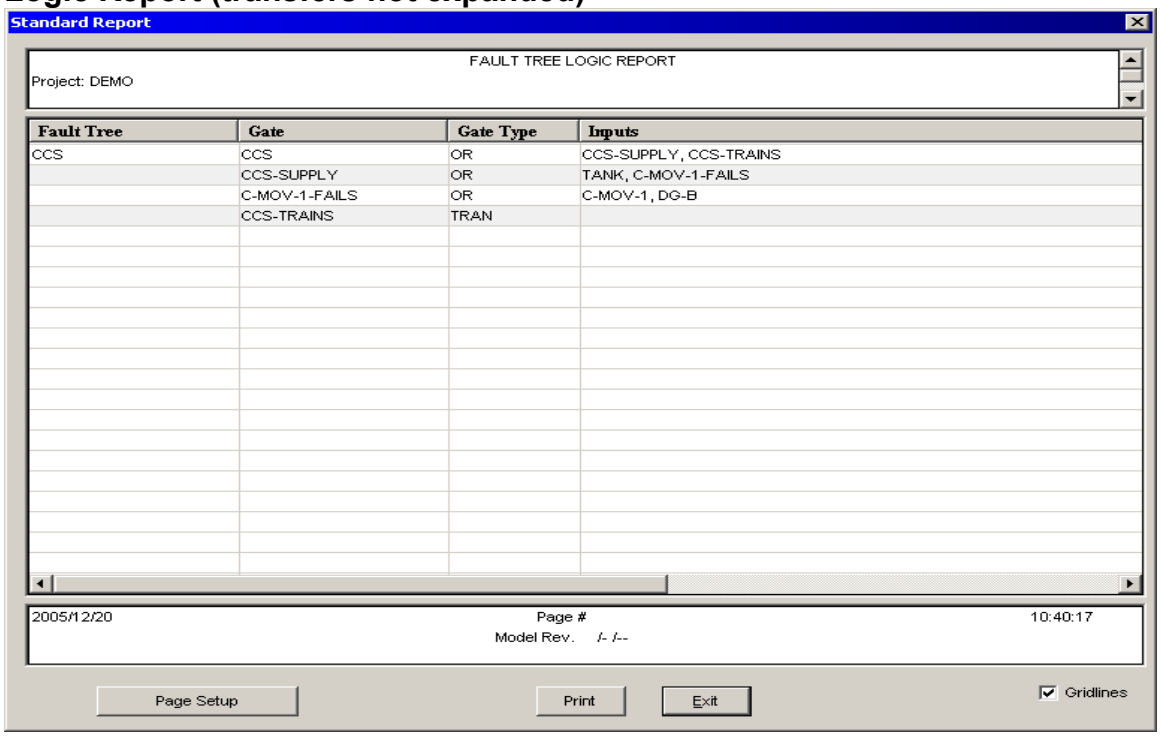
Fault Tree Base Case versus Current Case Report



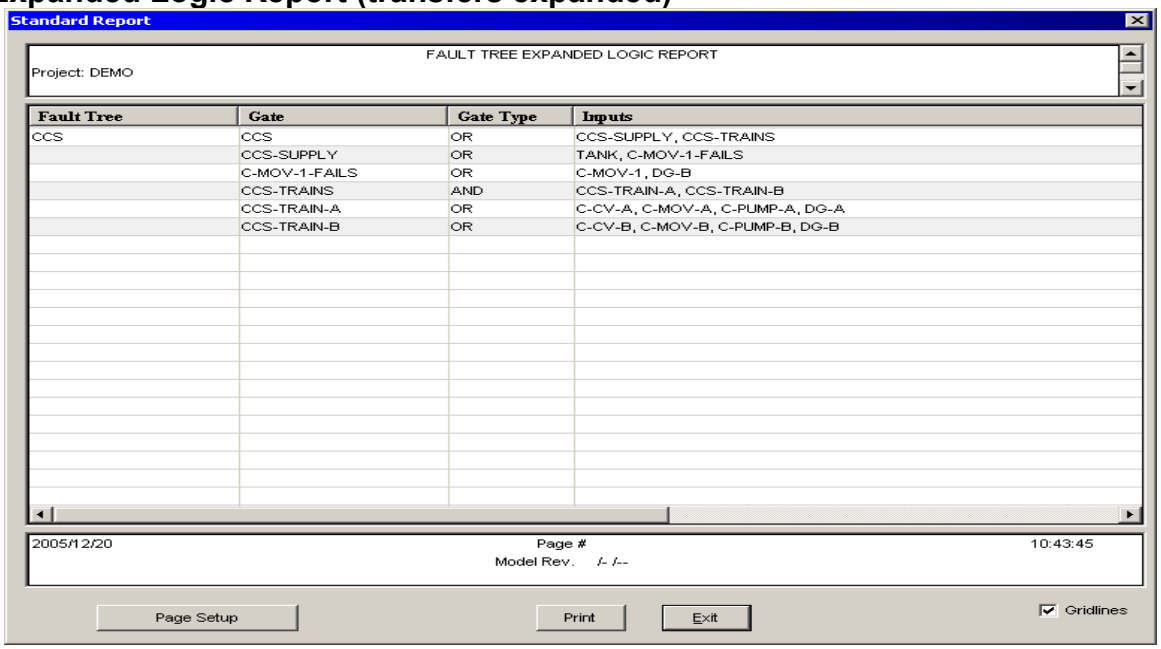
Fault Tree Logic Reports



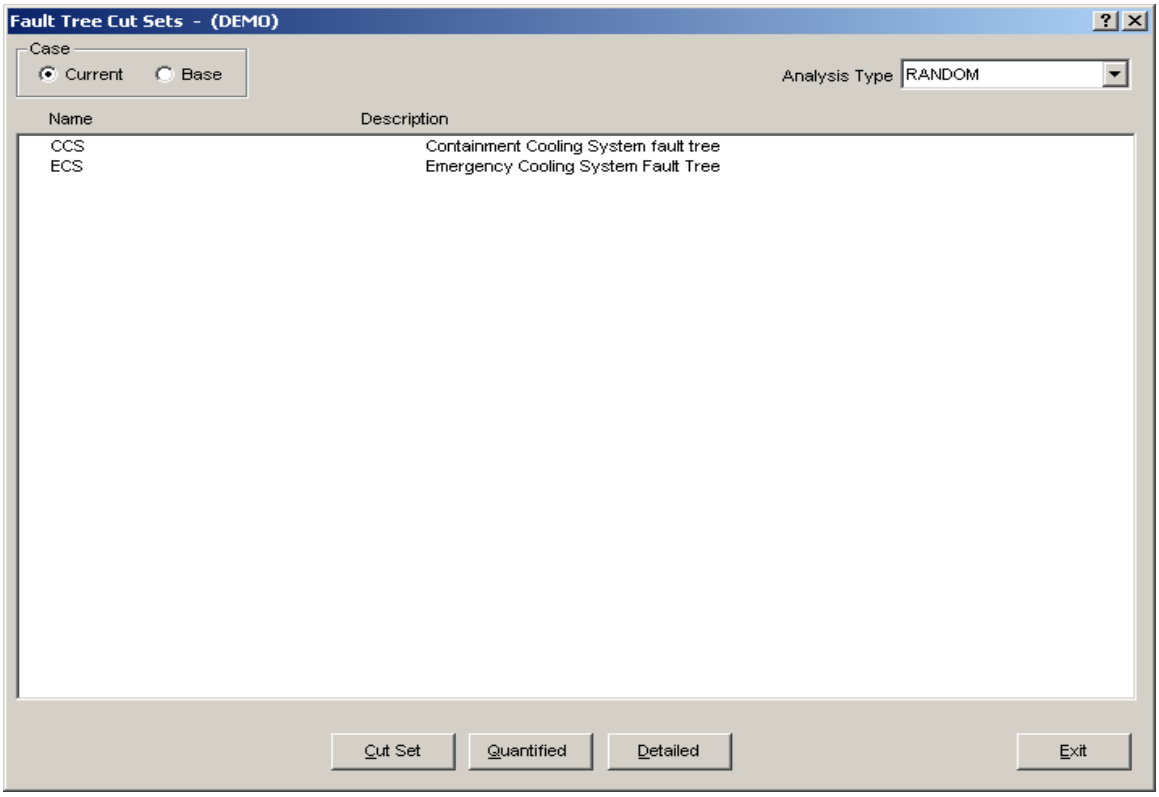
Logic Report (transfers not expanded)



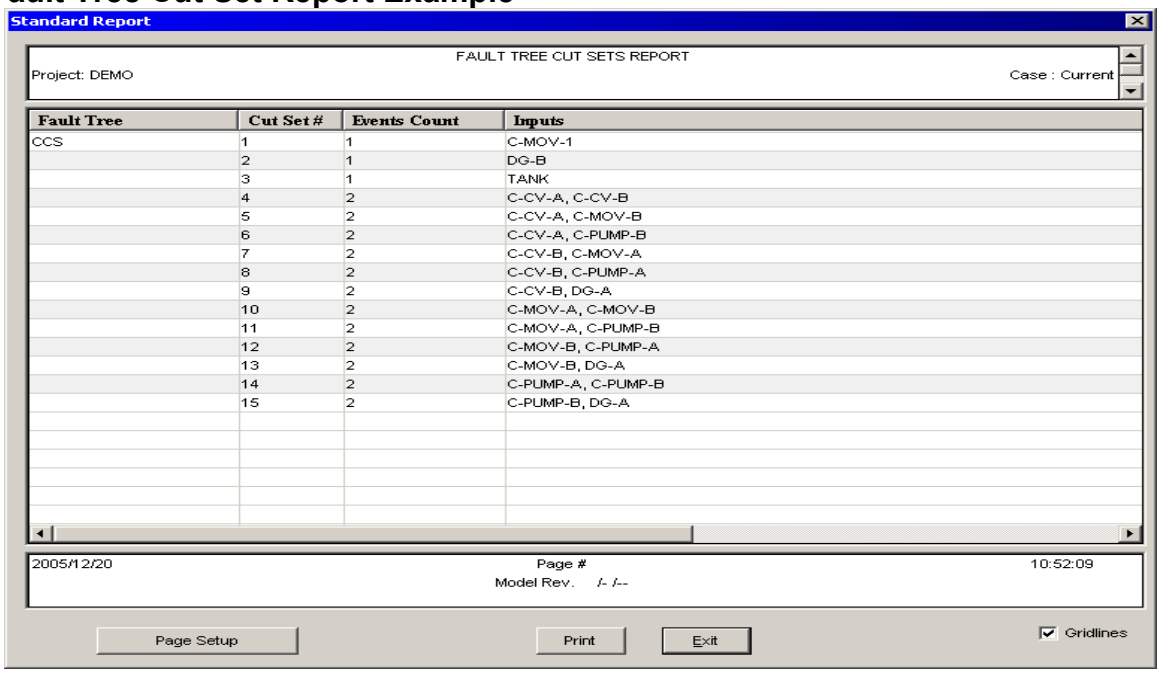
Expanded Logic Report (transfers expanded)



Fault Tree Cut Set Reports



Fault Tree Cut Set Report Example



Standard Report

FAULT TREE CUT SETS REPORT

Project: DEMO Case : Current

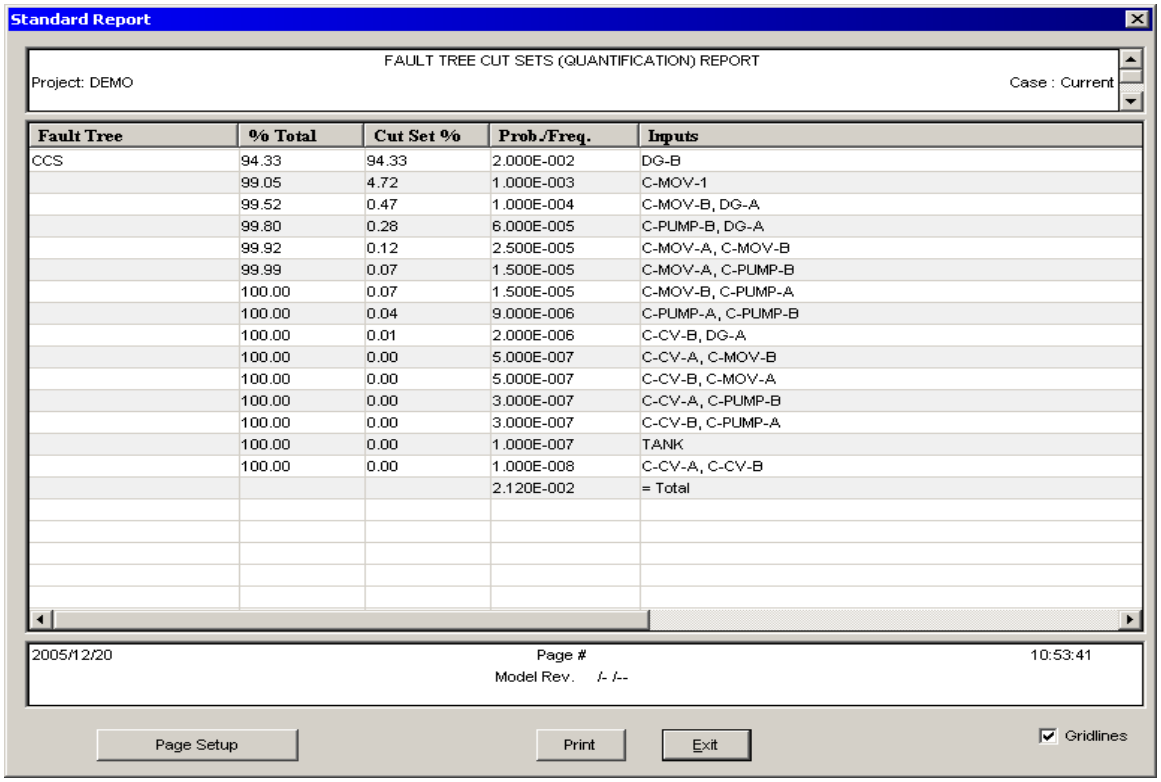
Fault Tree	Cut Set #	Events Count	Inputs
CCS	1	1	C-MOV-1
	2	1	DG-B
	3	1	TANK
	4	2	C-CV-A, C-CV-B
	5	2	C-CV-A, C-MOV-B
	6	2	C-CV-A, C-PUMP-B
	7	2	C-CV-B, C-MOV-A
	8	2	C-CV-B, C-PUMP-A
	9	2	C-CV-B, DG-A
	10	2	C-MOV-A, C-MOV-B
	11	2	C-MOV-A, C-PUMP-B
	12	2	C-MOV-B, C-PUMP-A
	13	2	C-MOV-B, DG-A
	14	2	C-PUMP-A, C-PUMP-B
	15	2	C-PUMP-B, DG-A

2005/12/20 Page # 10:52:09

Model Rev. /- /--

Page Setup Print Exit Gridlines

Fault Tree Quantified Cut Set Report



Standard Report

FAULT TREE CUT SETS (QUANTIFICATION) REPORT

Project: DEMO Case : Current

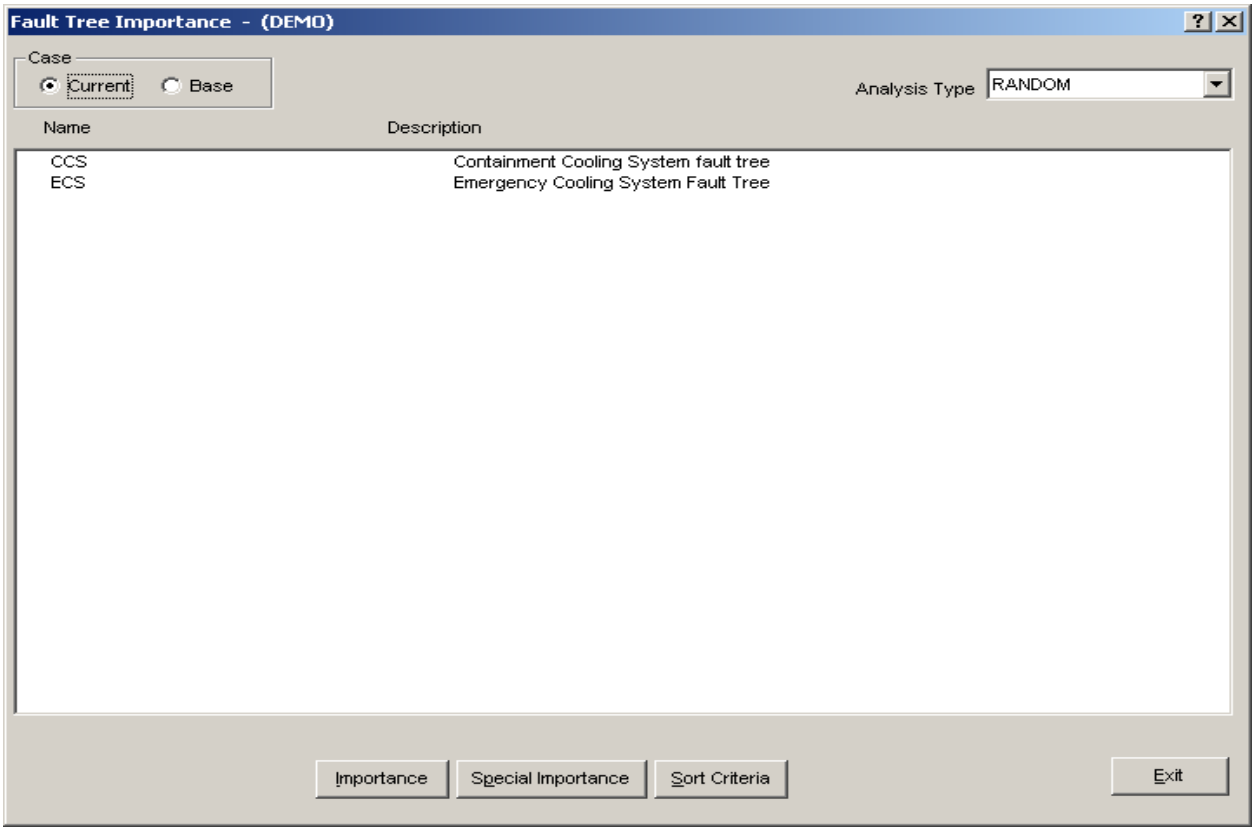
Fault Tree	% Total	Cut Set %	Prob./Freq.	Inputs
CCS	94.33	94.33	2.000E-002	DG-B
	99.05	4.72	1.000E-003	C-MOV-1
	99.52	0.47	1.000E-004	C-MOV-B, DG-A
	99.80	0.28	6.000E-005	C-PUMP-B, DG-A
	99.92	0.12	2.500E-005	C-MOV-A, C-MOV-B
	99.99	0.07	1.500E-005	C-MOV-A, C-PUMP-B
	100.00	0.07	1.500E-005	C-MOV-B, C-PUMP-A
	100.00	0.04	9.000E-006	C-PUMP-A, C-PUMP-B
	100.00	0.01	2.000E-006	C-CV-B, DG-A
	100.00	0.00	5.000E-007	C-CV-A, C-MOV-B
	100.00	0.00	5.000E-007	C-CV-B, C-MOV-A
	100.00	0.00	3.000E-007	C-CV-A, C-PUMP-B
	100.00	0.00	3.000E-007	C-CV-B, C-PUMP-A
	100.00	0.00	1.000E-007	TANK
	100.00	0.00	1.000E-008	C-CV-A, C-CV-B
			2.120E-002	= Total

2005/12/20 Page # 10:53:41

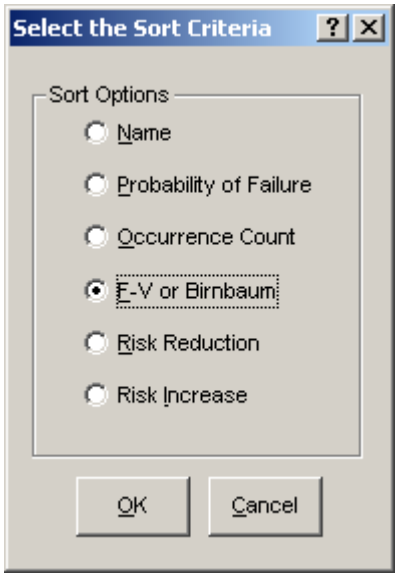
Model Rev. /- /--

Page Setup Print Exit Gridlines

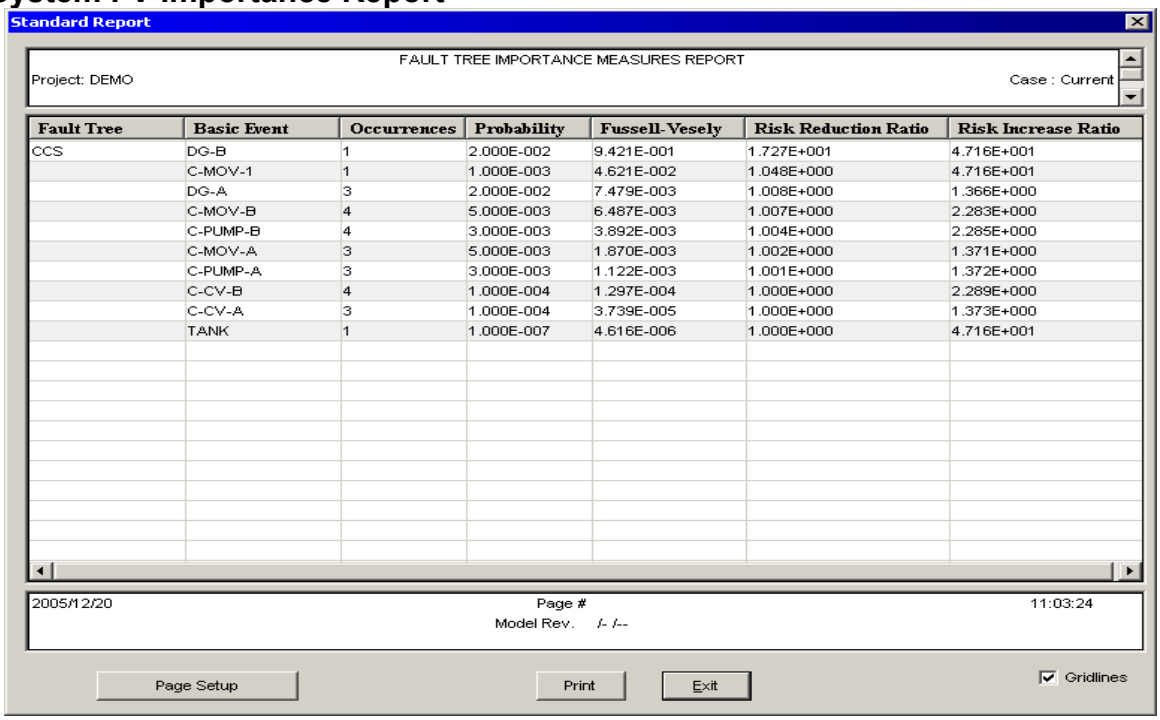
Fault Tree Importance Reports



Fault Tree Importance Sort (obtained by selecting "Sort Criteria")



System FV Importance Report



Standard Report

FAULT TREE IMPORTANCE MEASURES REPORT

Project: DEMO Case : Current

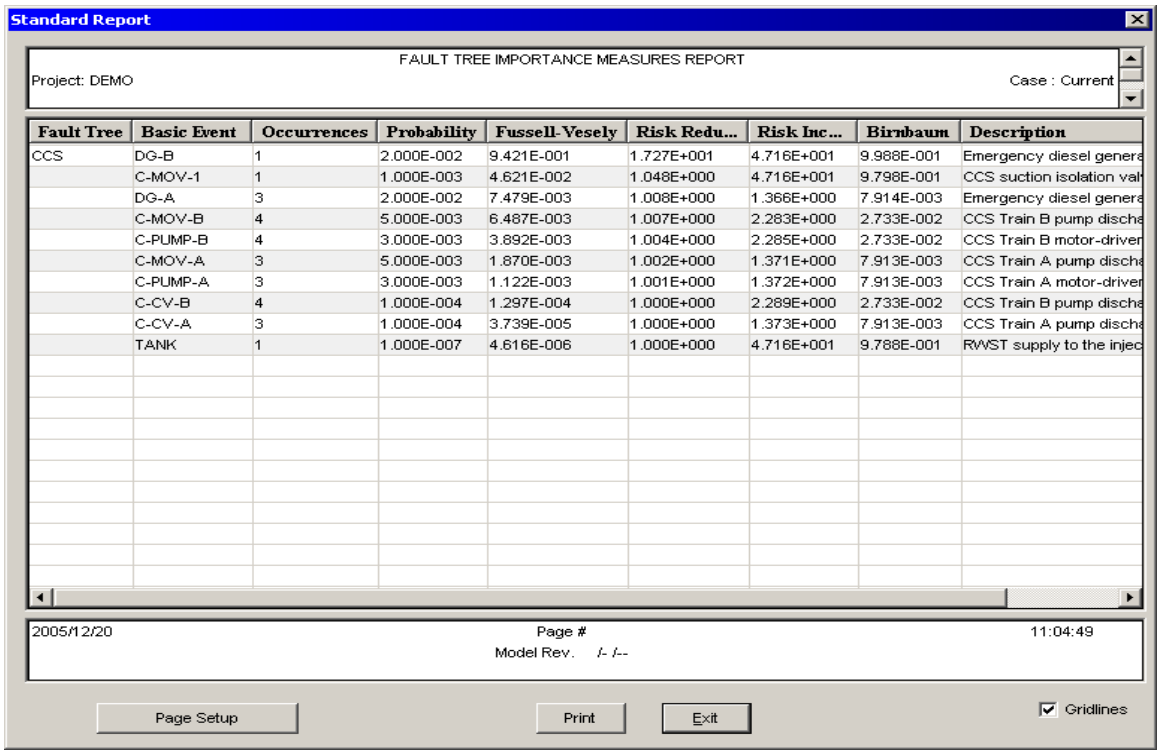
Fault Tree	Basic Event	Occurrences	Probability	Fussell-Vesely	Risk Reduction Ratio	Risk Increase Ratio
CCS	DG-B	1	2.000E-002	9.421E-001	1.727E+001	4.716E+001
	C-MOV-1	1	1.000E-003	4.621E-002	1.048E+000	4.716E+001
	DG-A	3	2.000E-002	7.479E-003	1.008E+000	1.366E+000
	C-MOV-B	4	5.000E-003	6.487E-003	1.007E+000	2.283E+000
	C-PUMP-B	4	3.000E-003	3.892E-003	1.004E+000	2.285E+000
	C-MOV-A	3	5.000E-003	1.870E-003	1.002E+000	1.371E+000
	C-PUMP-A	3	3.000E-003	1.122E-003	1.001E+000	1.372E+000
	C-CV-B	4	1.000E-004	1.297E-004	1.000E+000	2.289E+000
	C-CV-A	3	1.000E-004	3.739E-005	1.000E+000	1.373E+000
	TANK	1	1.000E-007	4.616E-006	1.000E+000	4.716E+001

2005/12/20 Page # 11:03:24

Model Rev. /- /--

Page Setup Print Exit Gridlines

Special Importance Report



Standard Report

FAULT TREE IMPORTANCE MEASURES REPORT

Project: DEMO Case : Current

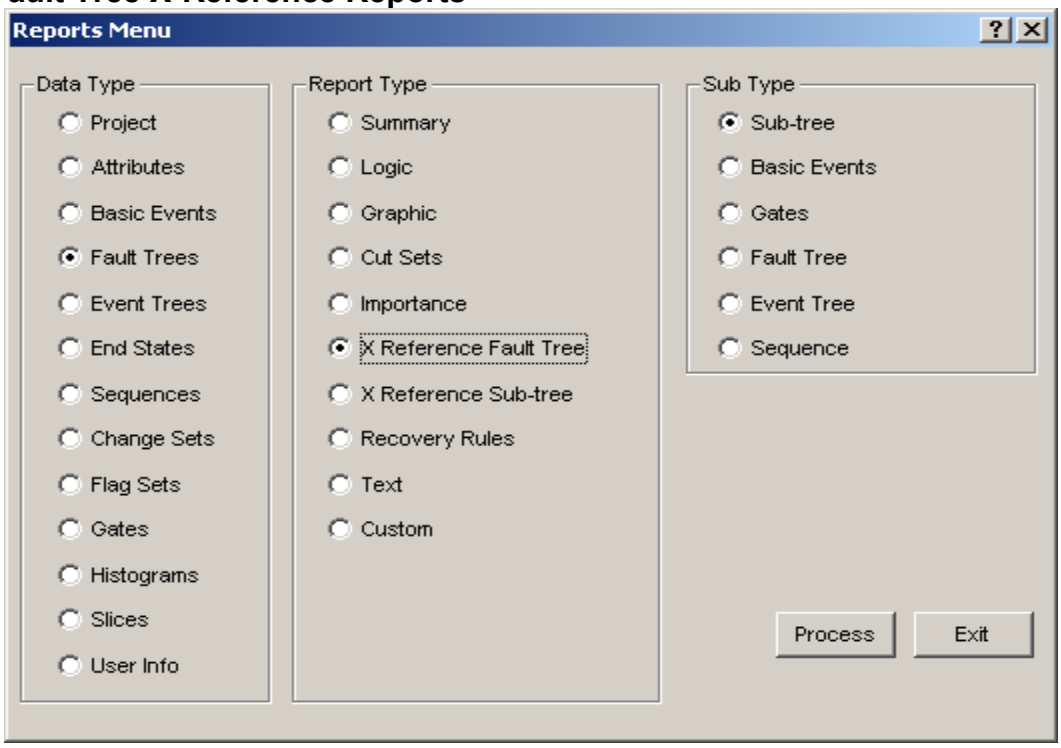
Fault Tree	Basic Event	Occurrences	Probability	Fussell-Vesely	Risk Redu...	Risk Inc...	Birnbaum	Description
CCS	DG-B	1	2.000E-002	9.421E-001	1.727E+001	4.716E+001	9.988E-001	Emergency diesel genera
	C-MOV-1	1	1.000E-003	4.621E-002	1.048E+000	4.716E+001	9.798E-001	CCS suction isolation val
	DG-A	3	2.000E-002	7.479E-003	1.008E+000	1.366E+000	7.914E-003	Emergency diesel genera
	C-MOV-B	4	5.000E-003	6.487E-003	1.007E+000	2.283E+000	2.733E-002	CCS Train B pump dische
	C-PUMP-B	4	3.000E-003	3.892E-003	1.004E+000	2.285E+000	2.733E-002	CCS Train B motor-driver
	C-MOV-A	3	5.000E-003	1.870E-003	1.002E+000	1.371E+000	7.913E-003	CCS Train A pump dische
	C-PUMP-A	3	3.000E-003	1.122E-003	1.001E+000	1.372E+000	7.913E-003	CCS Train A motor-driver
	C-CV-B	4	1.000E-004	1.297E-004	1.000E+000	2.289E+000	2.733E-002	CCS Train B pump dische
	C-CV-A	3	1.000E-004	3.739E-005	1.000E+000	1.373E+000	7.913E-003	CCS Train A pump dische
	TANK	1	1.000E-007	4.616E-006	1.000E+000	4.716E+001	9.788E-001	RWST supply to the injec

2005/12/20 Page # 11:04:49

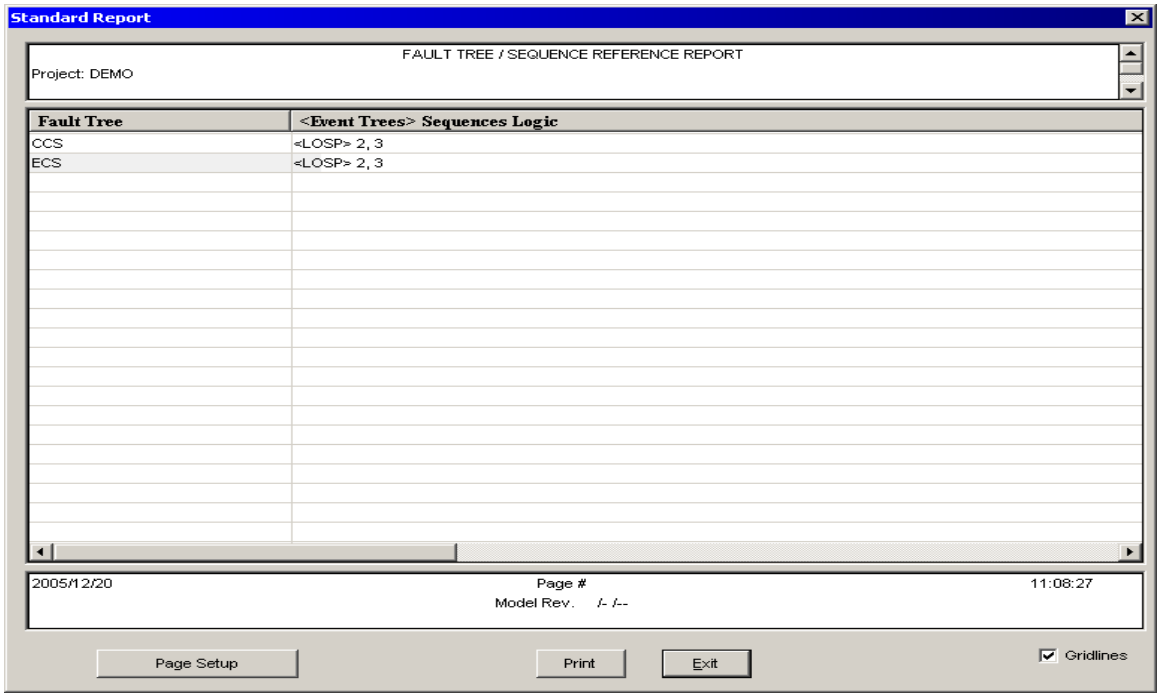
Model Rev. /- /--

Page Setup Print Exit Gridlines

Fault Tree X-Reference Reports



Fault Tree - Sequence X-Reference Report

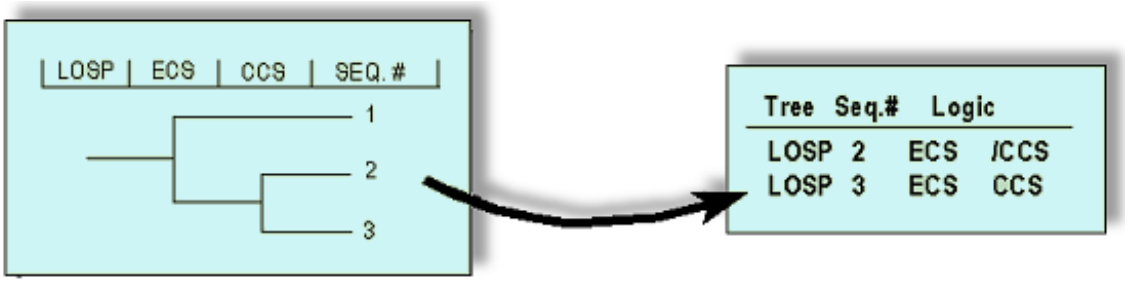


11 Linking Event Tree Sequences

Section 11 describes SAPHIRE event tree linking and process required to link event tree sequences.

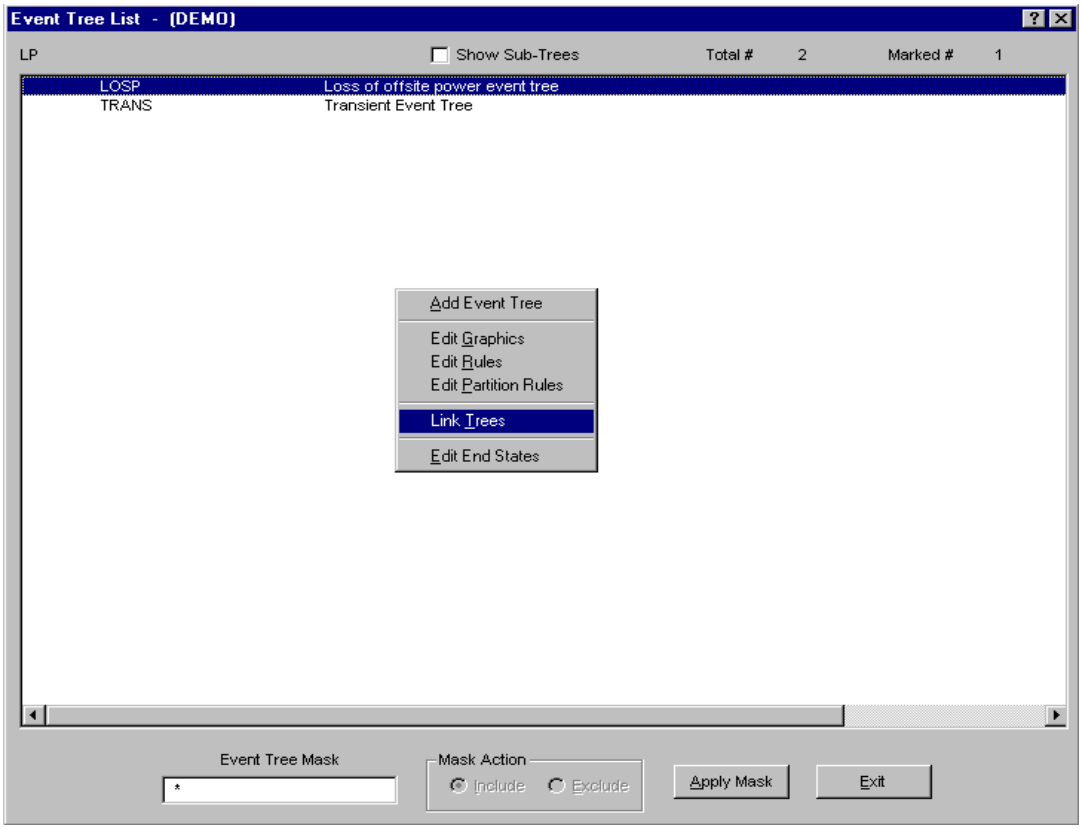
11.1. Linking Event Trees

"Linking" event trees is the process of generating sequence logic using the event tree graphical files.



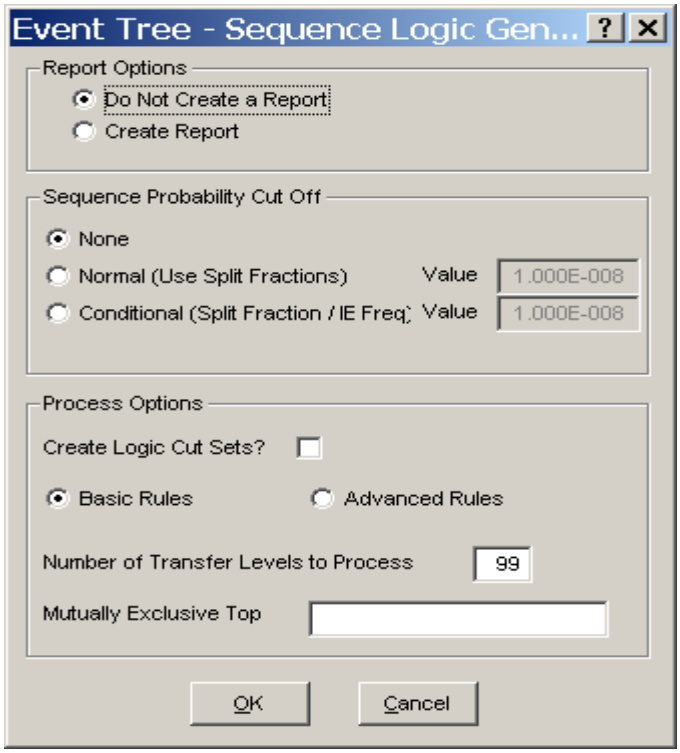
Menus and Options for Linking Event Tree Sequences

- Select **Event Tree** from the menu bar.
- The event trees are now displayed.
- Mark the event trees using the mask feature, or individually, using the mouse.



♦ Right-click to invoke the pop-up menu and select the **Link Trees** option.

Sequence Generation Information



- Enter the desired parameters on the dialog and choose the **OK** button to link event tree sequences.

  NOTE: Generally, the "sequence probability cut off" should be set to none.

Report Options

Do Not Create a Report

By selecting this radio button, the no report will be generated when SAPHIRE links the event tree.

Create Report

By selecting this radio button, SAPHIRE automatically creates a report of the sequence logic and the report is displayed, on screen. The report can now be printed to the default printer or sent to a file after choosing the file format.

Sequence Probability Cut Off

None

Generate sequence logic. This option is used for traditional event tree linking.

Normal (Use Split Fraction)

Generate sequence cut sets using the top events as split fractions. The sequence cut sets will be truncated based upon the cut off value specified. This truncation value will be used to truncate sequences where the value, including initiating event, are smaller than the specified value.

Conditional (Split Fraction/IE Frequency)

Generate sequence cut sets using the top events as split fractions. The sequence cut sets will be truncated based upon the cut off value specified. This truncation value will be used to truncate sequences where the value, based solely on the split fraction values, are smaller than the specified value.

Process Options

Create Logic Cut Sets

Will create a single cut set for the sequences if check box is checked. This cut set contains basic events representing each top event in the sequence. If not checked, then SAPHIRE will use the fault trees to generate cut sets (the traditional approach).

Number of Transfer Levels to Process
A level is a transfer to another event tree (subtree). The default (99) will generate sequences for all subtrees. If the level is specified as less than 99, only that number of subtrees will be processed.

Mutually Exclusive Top Name
Generally left blank. This allows you to specify a top event (associated with a fault tree) that will be added to each sequence as a success event. The top will appear in the logic as a complemented system and will be treated accordingly when the sequence is solved. This method has been superseded by the link rules option.

11.2. Reporting Event Tree Sequence Logic

You can view the event tree sequence logic in several ways:

- ◊ By selecting the **Create Report** radio button on the Sequence Logic Generate Options dialog, the logic is sent to the screen, from which the logic can be printed to the default printer or sent to a text file. The text file can be viewed after you exit SAPHIRE and use a text editor to open the file.

- ◊ Select **Report** from the menu bar, then select the **Sequences ➔ Logic** radio buttons to preview, print or create a file containing a report of the sequence logic.

SAPHIRE automatic sequence naming will add a "-" to the sequence identifier each time an event tree transfer is encountered.

12 Generating Event Tree Cut Sets

Section 12 describes how to generate event tree cut sets. Model preparation prior to generating cut sets is discussed, and the various analysis and truncation options are described. Cut set display features are also presented.

12.1. Prerequisites for Generating Event Tree Cut Sets

1. Event tree logic was created by using the event tree graphical editor.

2. Event tree logic was "linked" using **Event Tree ➔ Link Trees**.

3. Basic event data was added through the **Modify ➔ Basic Events** option.

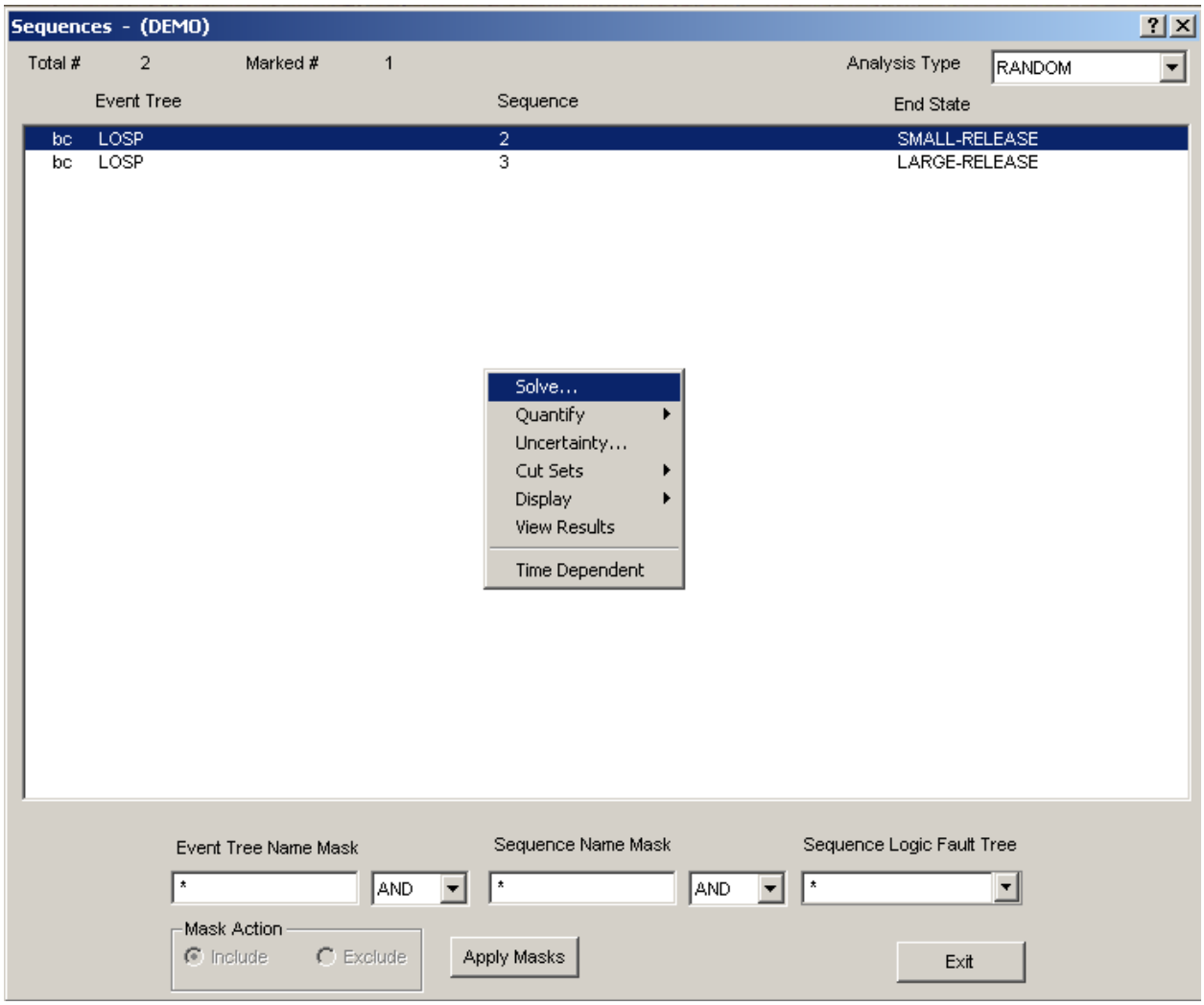
4. Basic event data was prepared for model processing by using the **Generate** option.

   *Note: Fault tree cut sets **do not** need to be generated prior to generating sequence cut sets.*

12.2. Menus and Options for Event Tree Cut Set Generation

- Select **Sequence** from the menu bar.

- Mark the sequences using the mask feature, or mark sequences individually using the mouse.



♦ Right-click to invoke the pop-up menu and select the **Solve** option.

b - flags sequences with existing base case cut sets

c - flags sequences with existing current case cut sets

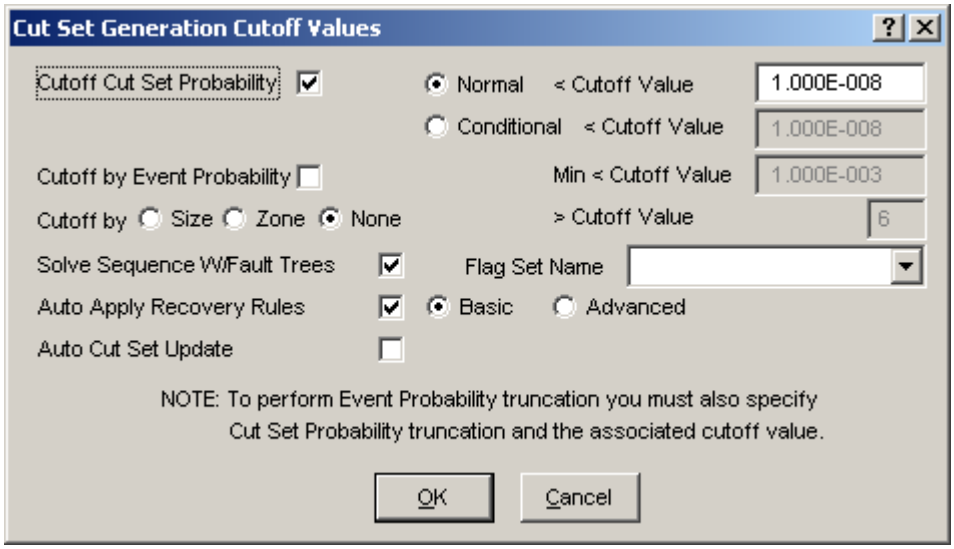
Analysis Type

Select the RANDOM analysis type for normal PRA analysis.

Solve

This option uses the event tree logic and fault trees associated with event tree top events.

Cut Set Generation Cutoff Values



Enter the desired truncation parameters, and choose the **OK** button to begin generating cut sets.

Cutoff by Cut Set Probability – If you check this box, then those cut sets below the value in the ***< Cutoff Value*** field will not be retained.

> Note: There are two different types of cutoff cut set probability.
>
> "Normal" uses the initiating event frequency in determining the cut off truncation value. This is the standard truncation option.
>
> "Conditional" ignores the initiating event value (i.e., assumes it is 1.0) thereby providing a conditional cutoff value (conditional upon the initiating event occurring).

Cutoff by Event Probability – Typically not used.

Cut Set Size Truncation – Typically not used.

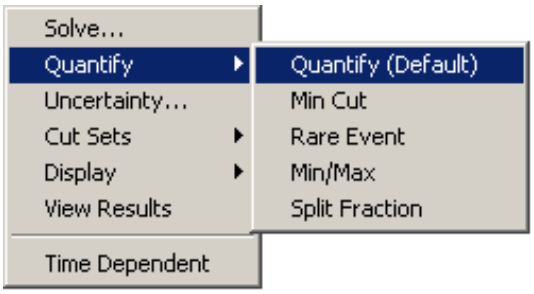
Solve Sequence with Fault Trees –If you check this box, then fault tree logic will be used. If the box is unchecked, previously generated fault tree cut sets will be used. Generally, this box will be checked.

Flag Set Name – A Flag Set containing House event settings may be specified in this field. If blank, the Flag Set specified in **Modify ➔ Event Trees ➔ Sequences ➔ Flags Name** will be used if a Flag Set was specified. Generally, this box is left blank.

Auto Apply Recovery Rules – If you check box, any recovery rules associated with the sequence(s) will automatically be applied after the sequence(s) cut sets have been generated. Generally, this box will be checked.

Auto Cut Set Update – If this box is checked, then the existing current case cut sets are reevaluated to remove all non-minimal cut sets and the fault tree probability is requantified.

Quantify



This sub-menu provides options for requantifying the sequence frequencies for existing current case cut sets. These options are designed to quickly requantify the cut sets when data changes have been made.

Split Fraction
Provides a minimal cut set upper bound *estimation* using the previously calculated system results for the failed or successful systems in the sequence. This option does not generate cut sets, and is only appropriate for event trees with independent top events. This option is generally not used.

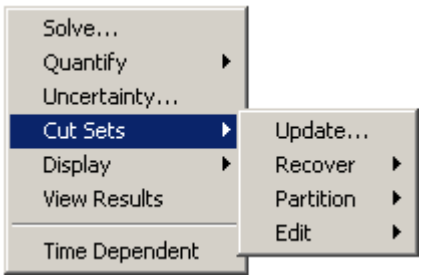
Min Max Quantification
The Min-Max quantification option quantifies the current case cut sets using an exact probability quantification algorithm.

see Section 7, Fault Tree Cut Set Generation

Uncertainty

Performs Monte Carlo or Latin Hypercube uncertainty analysis for the selected sequence individually or combined. Sequence uncertainty analysis is discussed further in Section 14.

Cut Sets



This sub-menu provides options for cut set manipulation.

Cut Set Update
This option uses the existing current case cut sets (unless the user specifies that base case cut sets are to be used instead). Non-minimal cut sets are eliminated and the sequence frequency is quantified using the minimal cut set upper bound.

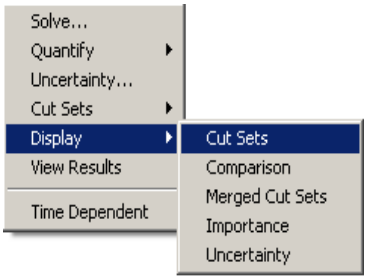
Recover Option
This option allows for post-processing rules to be created and applied to the accident sequence cut sets. The rules can be created and applied to individual sequences, event trees, or all sequences.

Partition Option
This option provides a mechanism to group sequence cut sets into user specified end states based upon partition rules.

Edit Option
This option provides the analyst a mechanism to edit the current case sequence cut sets.

12.3. Displaying Event Tree Cut Set Results

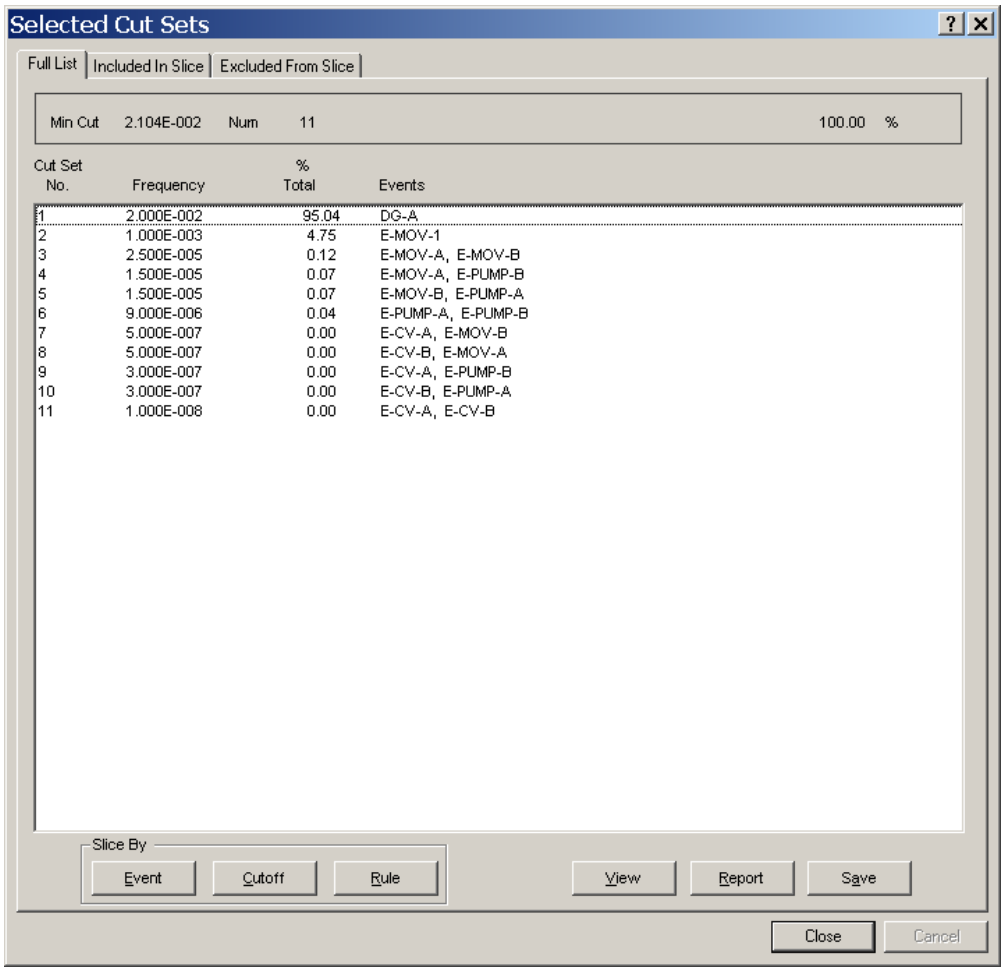
- To display sequence cut sets, select **Sequence** from the menu bar.
- Highlight the sequence that you want to view.
- Right-click to invoke the pop-up menu and select the **Display ➔ Cut Sets** option.



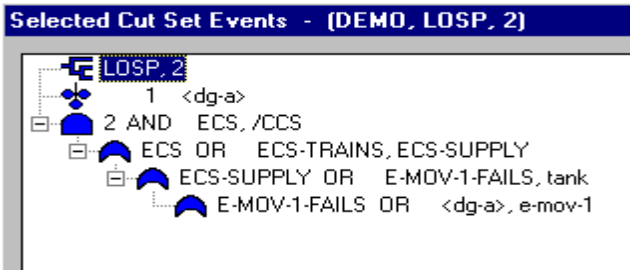
Display Sequence Cut Sets

- The sequence cut sets and minimal cut set upper bound approximation of the sequence frequency are now displayed.

- To view the basic events in a cut set, highlight the cut set, and click the **View** button.

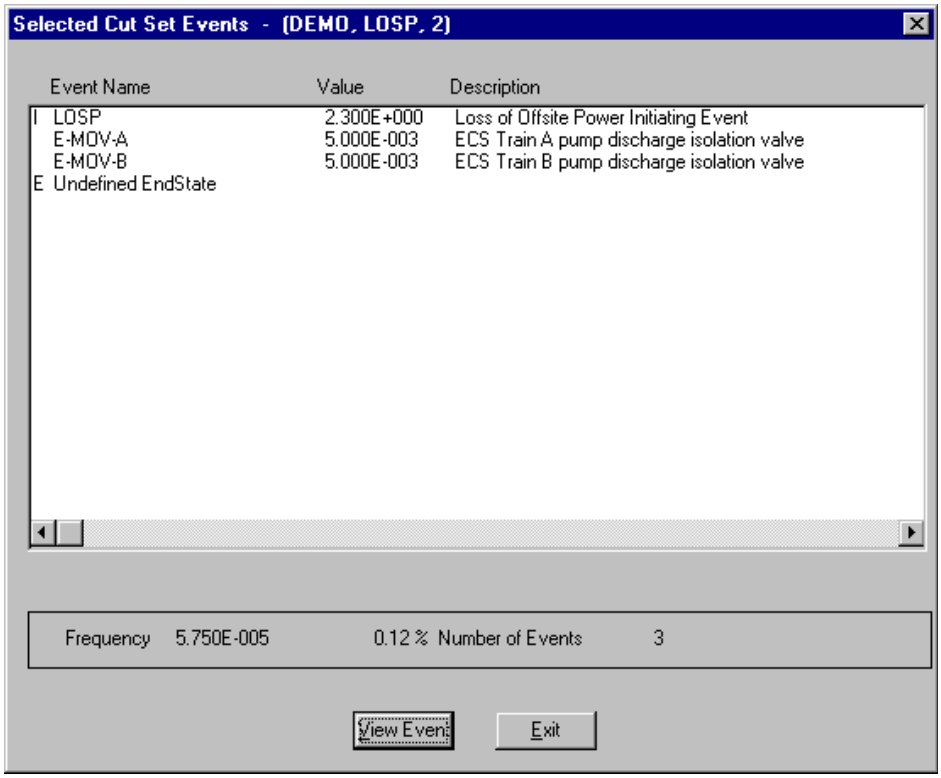


- To find out where the events in the cut set came from, right-click on a cut set and select the **Path** option.



Viewing Basic Event Information for an Individual Cut Set

- The basic events, their failure probabilities, and descriptions are now displayed.



- To view individual basic event information, highlight the event and choose the **View Event** button.

- To return to the list of cut sets, choose the **Exit** button.

12.4. Additional Event Tree Analysis Features

In complex PRA models, it may be necessary to use special features prior to cut set generation and/or after cut set generation. These features are addressed in the Advanced SAPHIRE course.

Prior to event tree sequence cut set generation:

Flag Sets may be used to set House events or Process Flags on a sequence-by-sequence basis.

After event tree sequence cut set generation:

Recover Cut Sets can be used as a rule-based automated way to add "recovery events" to the cut sets. These rules are a mechanism of post-processing the cut set list.

13 Sequence Uncertainty Analysis

Section 13 describes uncertainty analysis for event tree sequences. The concept of performing uncertainty analysis via Monte Carlo or Latin Hypercube sampling is discussed.

13.1. Sequence Uncertainty Analysis

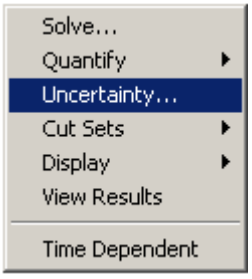
Uncertainty analysis calculates the variability of the sequence frequency resulting from uncertainties in the basic event probabilities and the initiating event frequency.

SAPHIRE provides two uncertainty analysis techniques:

- ◊ Simple Monte Carlo sampling
- ◊ Latin Hypercube sampling

13.2. Menus and Options for Performing Sequence Uncertainty Analysis

- ♦ Select **Sequence** from the menu bar.

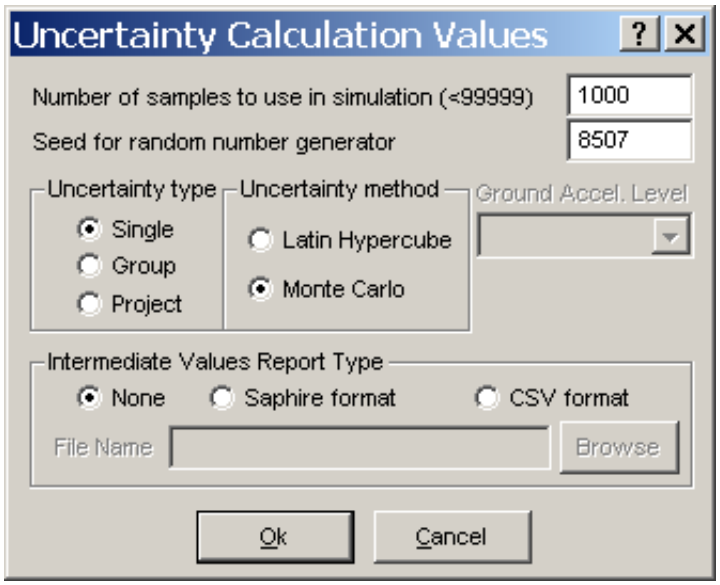


- ♦ Mark the sequence(s), right-click to invoke the pop-up menu.

- ♦ Select the **Uncertainty** option. The sequence uncertainty may be calculated for each sequence individually or as a group.

Uncertainty Calculation Values

- Enter the uncertainty calculation values on the dialog:



- Uncertainty results will be displayed briefly on the dialog following the uncertainty calculation.

Uncertainty Type

Specify whether the marked sequences are to be calculated individually (one at a time), as a group, or for the project (if all sequences are marked).

Uncertainty Method

Specify Monte Carlo or Latin Hypercube.

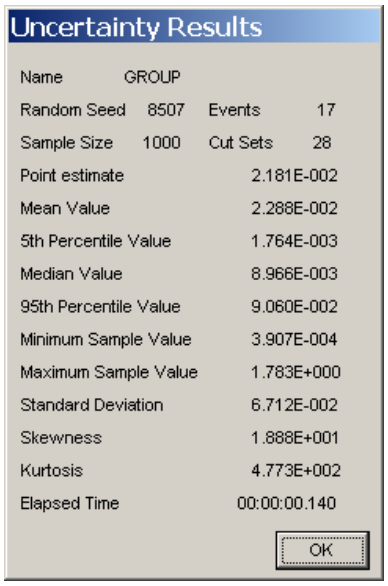
Number of Samples

Input the number of samples. (A larger number of samples will provide more accurate results but will require more time.)

Random Number Seed

Enter a value for the random number generator seed or accept the default. Enter zero to obtain a random seed from the system clock.

Uncertainty Results



- Uncertainty results will be displayed briefly on the screen following the uncertainty calculation.

- Uncertainty results for each sequence *when calculated individually* can be displayed:

  - Select **Sequence** from menu bar, and highlight the individual sequence

  - Right-click to invoke the pop-up menu and select the **Display** ➔ **Uncertainty** option.

- Uncertainty results for the *project* can be displayed:

  - Select **Sequence** from the menu bar, mark all the sequences in the family

  - Right-click to invoke the pop-up menu and select the **Display** ➔ **Uncertainty** option.

- Note: GROUP uncertainty results are displayed *only* on the screen (shown above). These results are not available from the **Display** option.

Notes

14 Sequence Importance Analysis

Section 14 describes the various sequence importance measures available in SAPHIRE.

14.1. Sequence Importance Measures

Importance measures provide “reliability-worth” information about basic events appearing in the cut sets of an event tree sequence.

- Components showing high relative importance may be candidates for either (1) close monitoring to ensure that the component does not degrade over time or (2) design changes to increase the component reliability.

  Ratio, Interval and Uncertainty importances can are calculated for the highlighted event tree sequences.

  - *Ratio Importance* – Fussell-Vesely Importance, Risk Reduction Ratio, and Risk Increase Ratio.

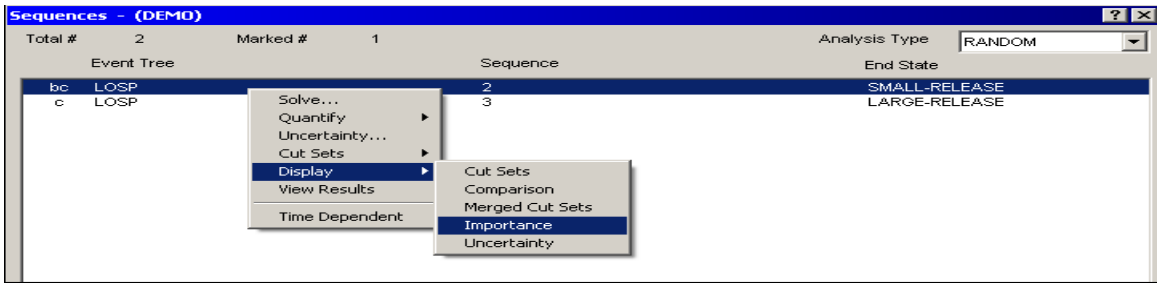
  - *Interval Importance* – Birnbaum Importance, Risk Reduction Interval, Risk Increase Interval.

  - *Uncertainty Importance* – This importance measure provides information about the uncertainty of the component (i.e., those components that will contribute the largest uncertainty to an uncertainty analysis).

[See Section 9.0 for a detailed equation description for the above importance measures]

14.2. Menus and Options for Calculating Sequence Importance Measures

♦ Select the **Sequence** from the menu bar.

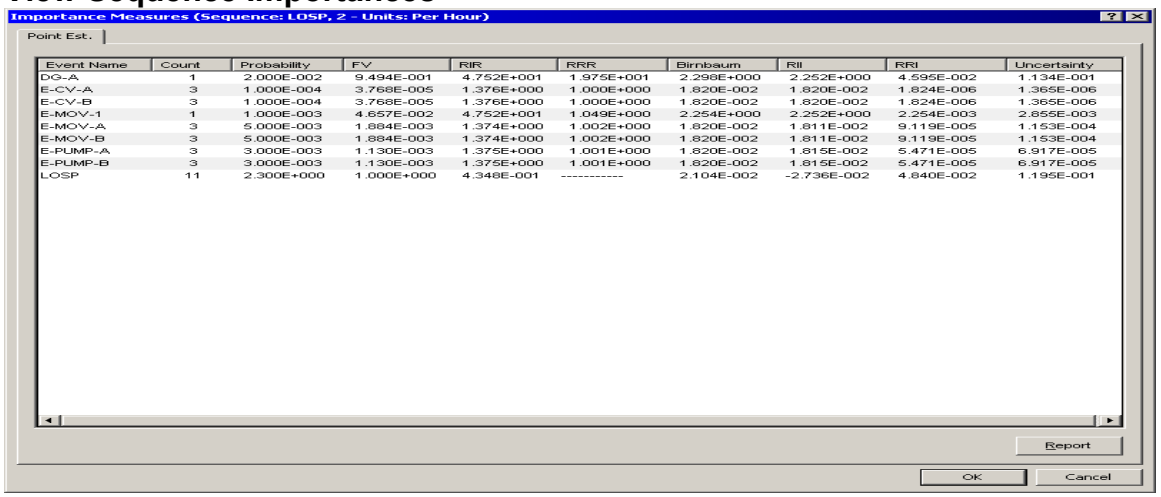


♦ Highlight an individual sequence, right-click to invoke the pop-up menu.

♦ Select **Importance**

♦ The importance measures are now displayed.

View Sequence Importances



Event Name	Count	Probability	FV	RIR	RRR	Birnbaum	RII	RRI	Uncertainty
DG-A	1	2.000E-002	9.494E-001	4.752E+001	1.975E+001	2.298E+000	2.252E+000	4.595E-002	1.134E-001
E-CV-A	3	1.000E-004	3.768E-005	1.376E+000	1.000E+000	1.820E-002	1.820E-002	1.824E-006	1.365E-006
E-CV-B	3	1.000E-004	3.768E-005	1.376E+000	1.000E+000	1.820E-002	1.820E-002	1.824E-006	1.365E-006
E-MOV-1	1	1.000E-003	4.657E-002	4.752E+001	1.049E+000	2.254E+000	2.252E+000	2.254E-003	2.855E-003
E-MOV-A	3	5.000E-003	1.884E-003	1.374E+000	1.002E+000	1.820E-002	1.811E-002	9.119E-005	1.153E-004
E-MOV-B	3	5.000E-003	1.884E-003	1.374E+000	1.002E+000	1.820E-002	1.811E-002	9.119E-005	1.153E-004
E-PUMP-A	3	3.000E-003	1.130E-003	1.375E+000	1.001E+000	1.820E-002	1.815E-002	5.471E-005	6.917E-005
E-PUMP-B	3	3.000E-003	1.130E-003	1.375E+000	1.001E+000	1.820E-002	1.815E-002	5.471E-005	6.917E-005
LOSP	11	2.300E+000	1.000E+000	4.348E-001	----------	2.104E-002	-2.736E-002	4.840E-002	1.195E-001

♦ To analyze the importance of a group of sequences, highlight the desired sequences, then select **Importance**.

♦ To print the importance measures either to a printer or a file, select the **Report** button.

♦ To sort by a column heading, click on the desired column heading.

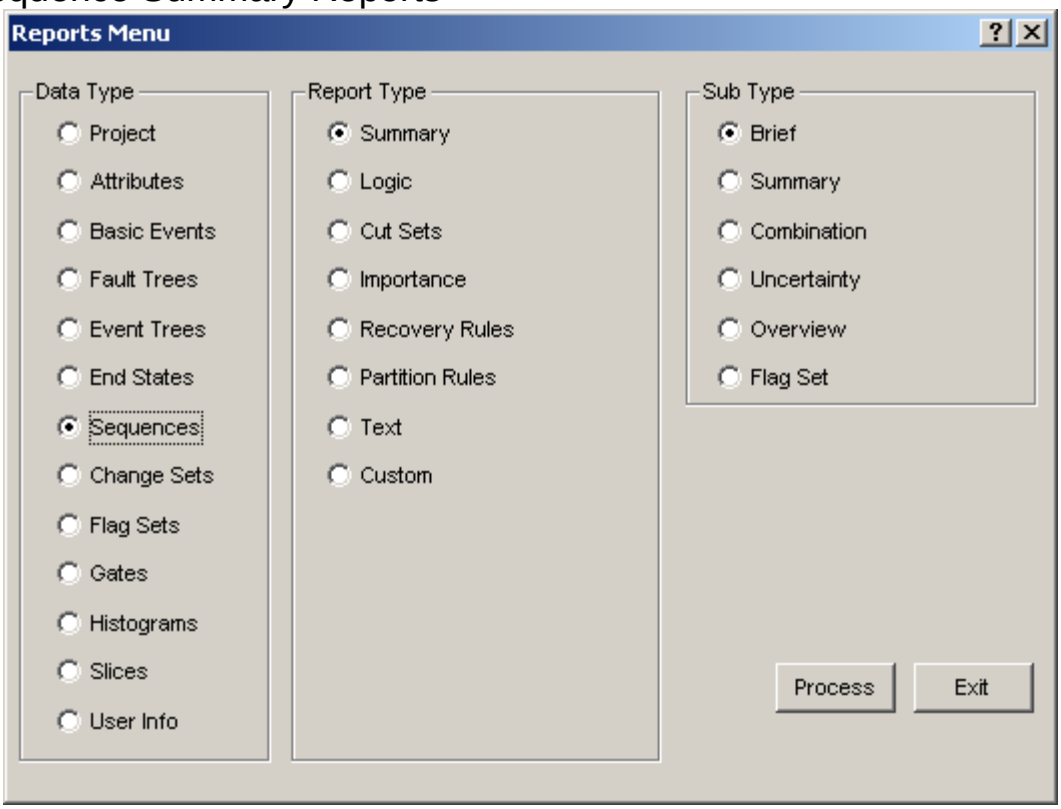
15 Reporting Sequence Results

Section 15 describes how to generate reports of sequence cut set results.

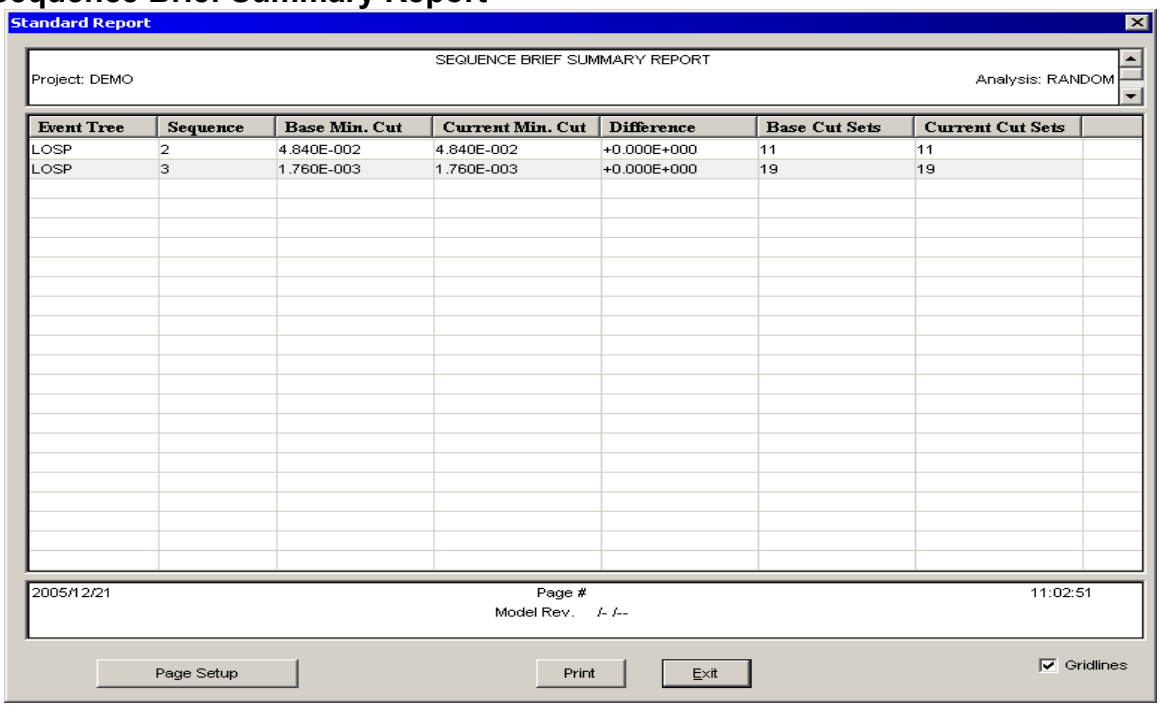
15.1. Sequence Report Menus and Examples

- To report sequence information, select **Report** from the menu bar.
- Select the **Sequence** radio button and then you can choose from several different report options.

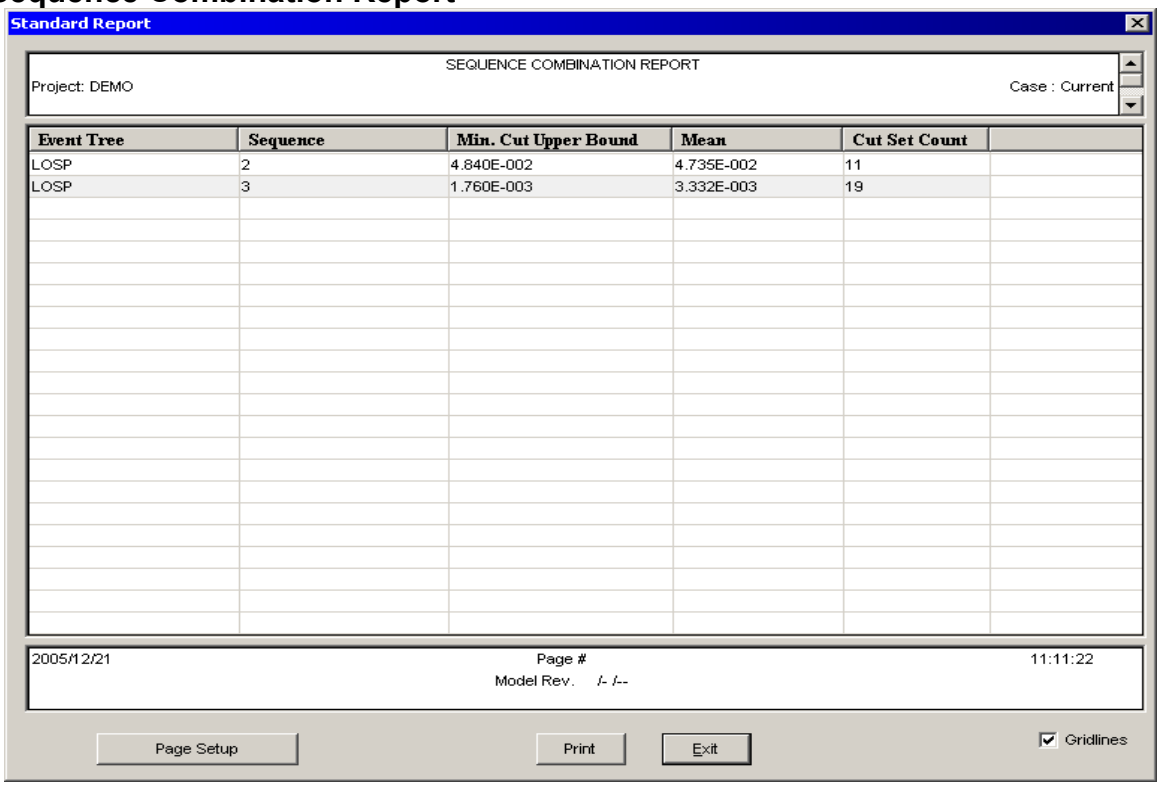
Sequence Summary Reports



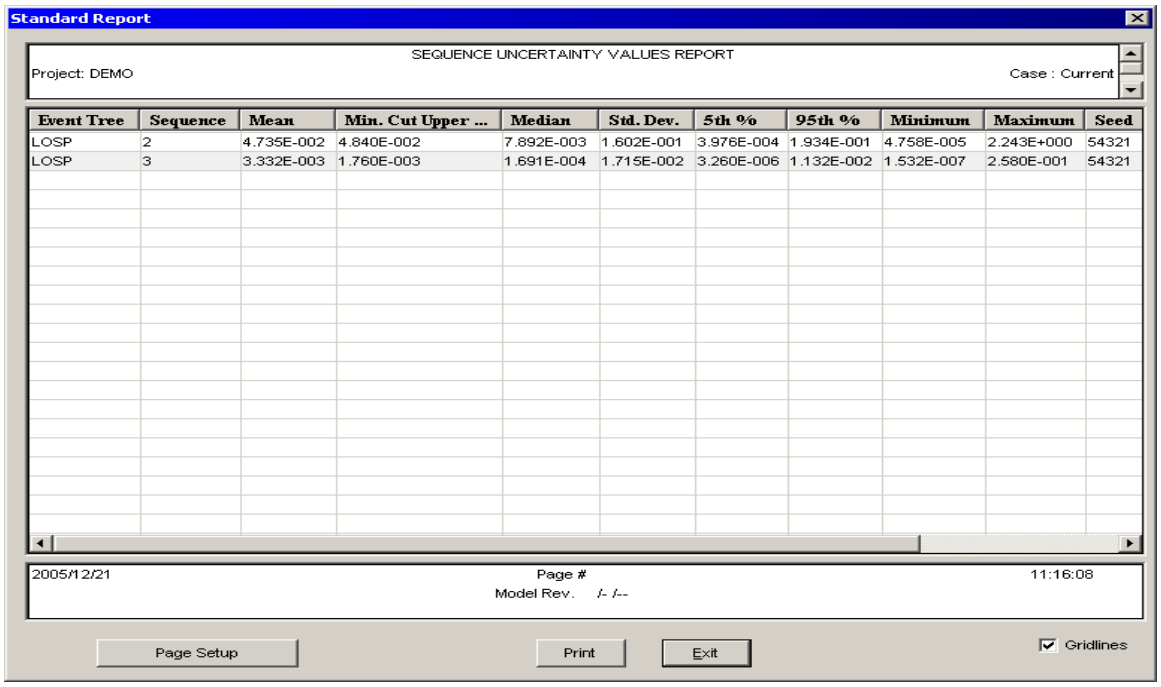
Sequence Brief Summary Report



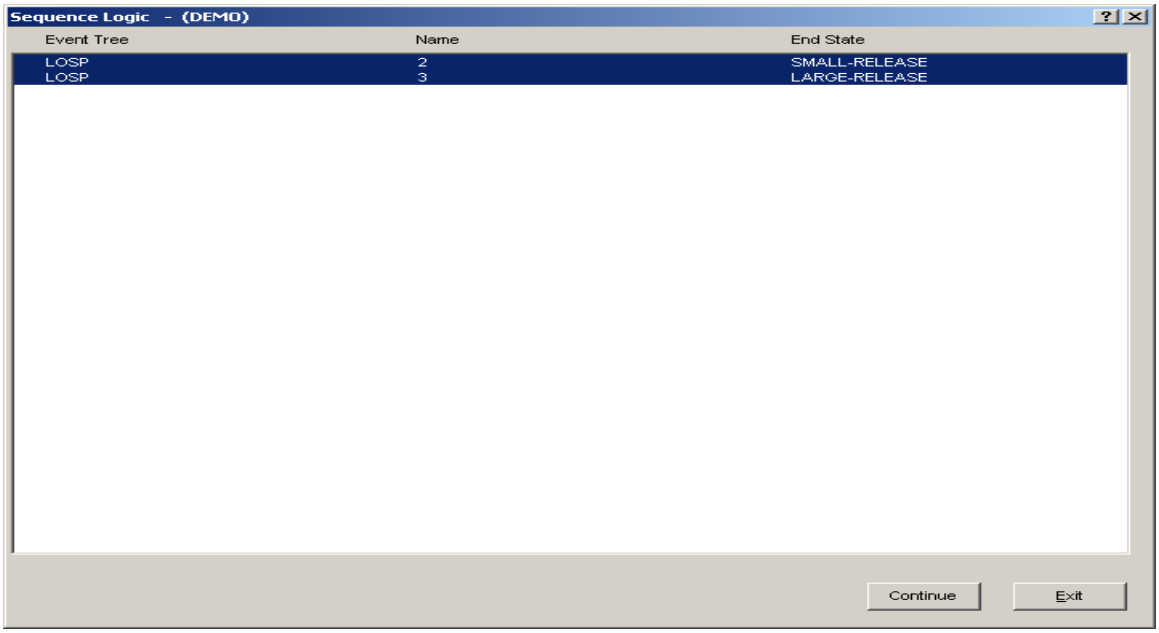
Sequence Combination Report



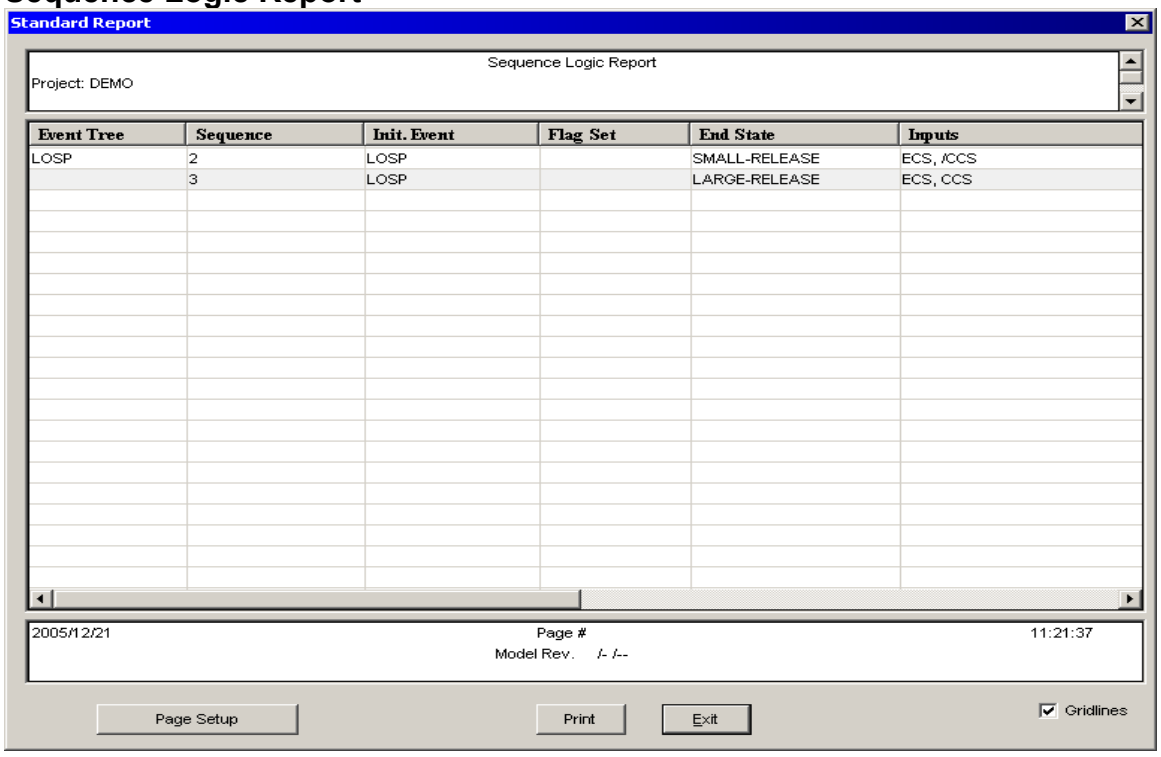
Sequence Uncertainty Values Report



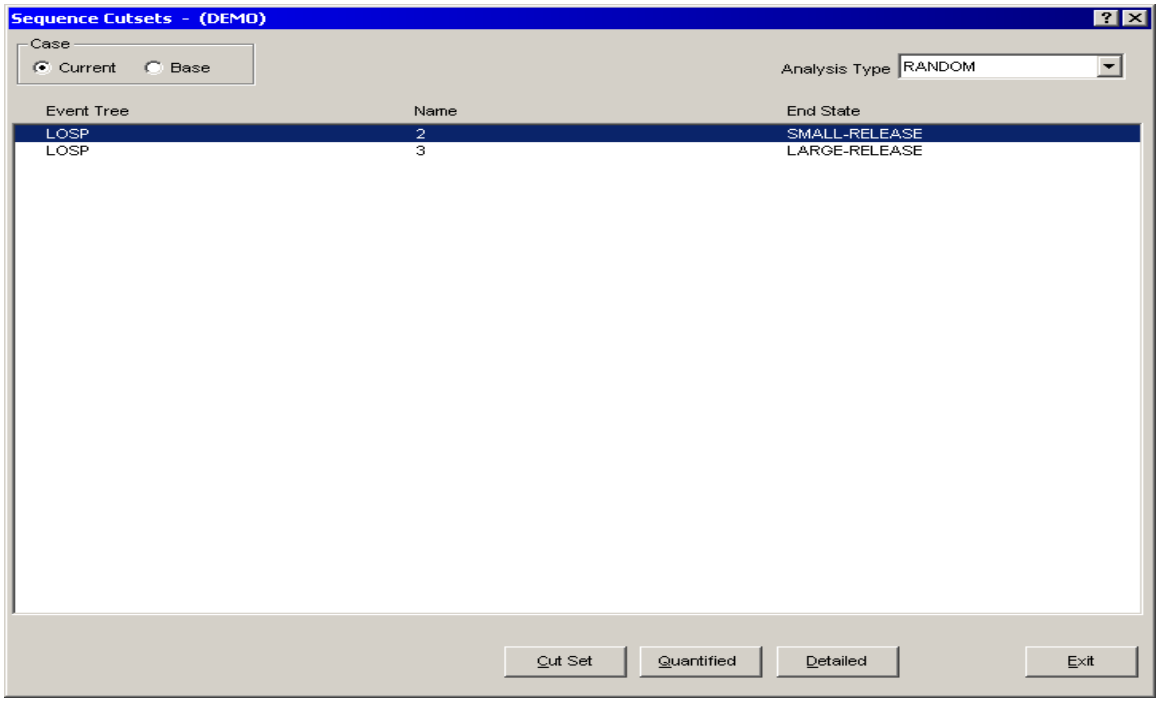
Sequence Logic Reports



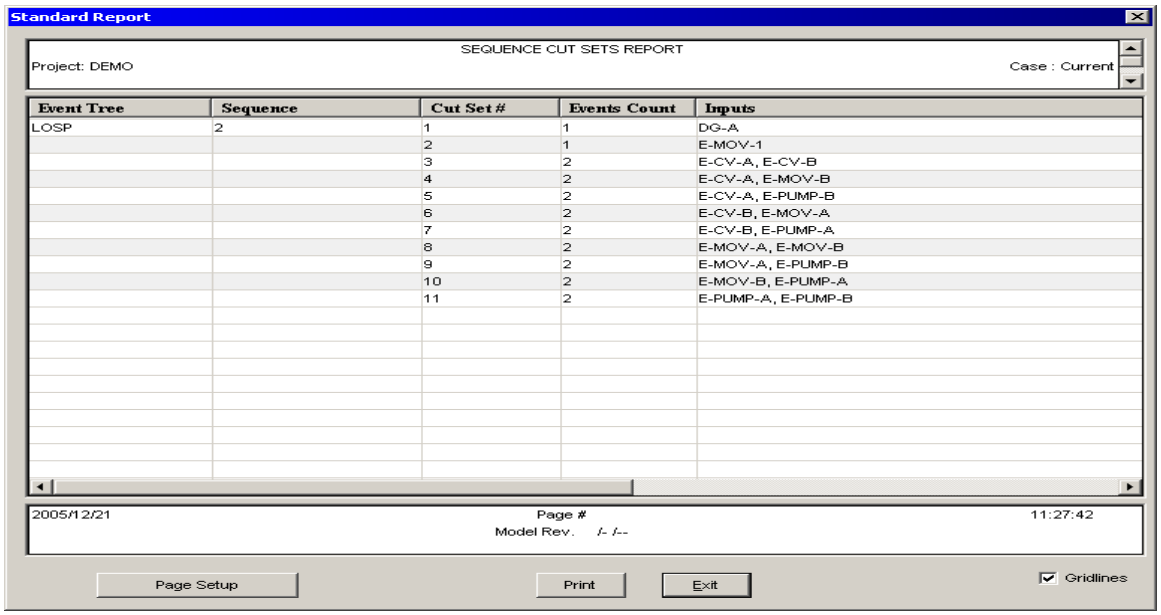
Sequence Logic Report



Sequence Cut Set Reports



Sequence Cut Set Report



Standard Report

SEQUENCE CUT SETS REPORT

Project: DEMO Case : Current

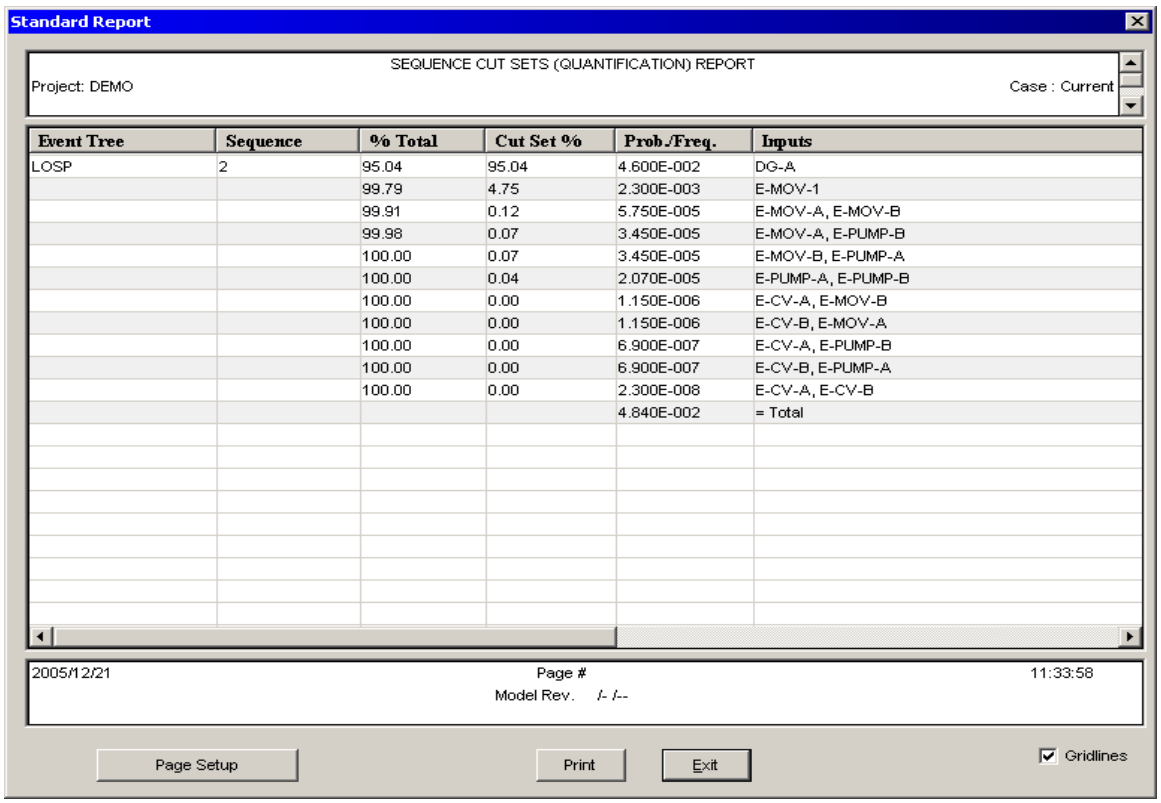
Event Tree	Sequence	Cut Set #	Events Count	Inputs
LOSP	2	1	1	DG-A
		2	1	E-MOV-1
		3	2	E-CV-A, E-CV-B
		4	2	E-CV-A, E-MOV-B
		5	2	E-CV-A, E-PUMP-B
		6	2	E-CV-B, E-MOV-A
		7	2	E-CV-B, E-PUMP-A
		8	2	E-MOV-A, E-MOV-B
		9	2	E-MOV-A, E-PUMP-B
		10	2	E-MOV-B, E-PUMP-A
		11	2	E-PUMP-A, E-PUMP-B

2005/12/21 Page # 11:27:42

Model Rev. /- /--

Page Setup Print Exit Gridlines

Sequence Quantified Cut Set Report



Standard Report

SEQUENCE CUT SETS (QUANTIFICATION) REPORT

Project: DEMO Case : Current

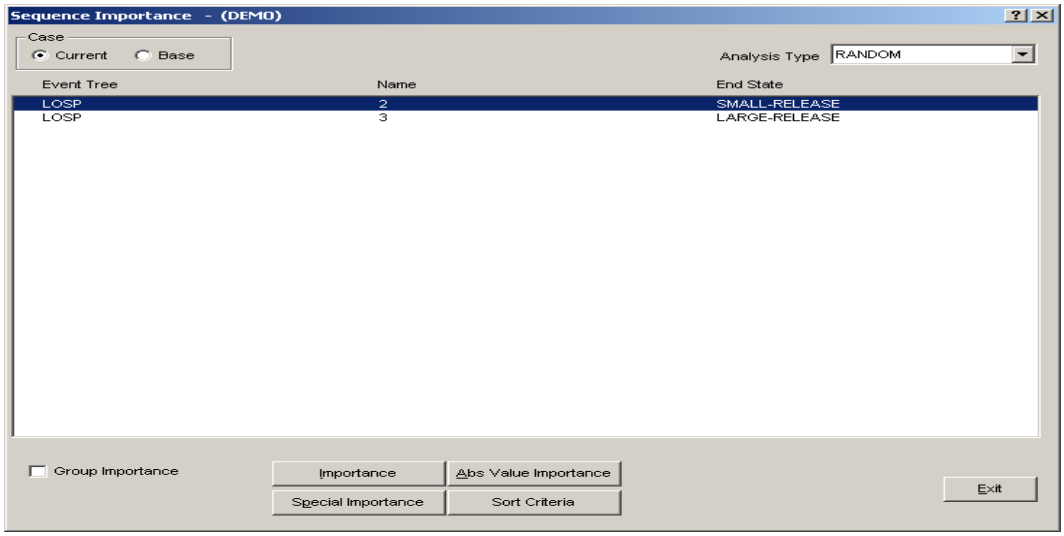
Event Tree	Sequence	% Total	Cut Set %	Prob./Freq.	Inputs
LOSP	2	95.04	95.04	4.600E-002	DG-A
		99.79	4.75	2.300E-003	E-MOV-1
		99.91	0.12	5.750E-005	E-MOV-A, E-MOV-B
		99.98	0.07	3.450E-005	E-MOV-A, E-PUMP-B
		100.00	0.07	3.450E-005	E-MOV-B, E-PUMP-A
		100.00	0.04	2.070E-005	E-PUMP-A, E-PUMP-B
		100.00	0.00	1.150E-006	E-CV-A, E-MOV-B
		100.00	0.00	1.150E-006	E-CV-B, E-MOV-A
		100.00	0.00	6.900E-007	E-CV-A, E-PUMP-B
		100.00	0.00	6.900E-007	E-CV-B, E-PUMP-A
		100.00	0.00	2.300E-008	E-CV-A, E-CV-B
				4.840E-002	= Total

2005/12/21 Page # 11:33:58

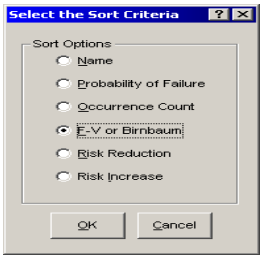
Model Rev. /- /--

Page Setup Print Exit Gridlines

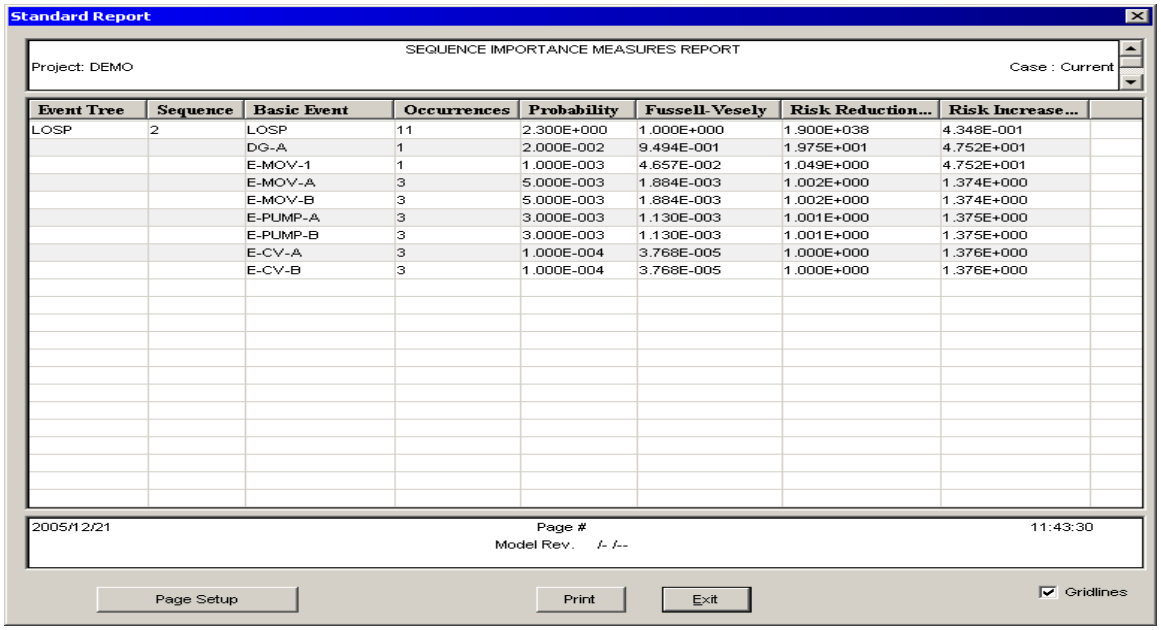
Sequence Importance Reports



In this example the “F-V or Birnbaum” sort option is selected.



Sequence Importance Measures Report



Event Tree	Sequence	Basic Event	Occurrences	Probability	Fussell-Vesely	Risk Reduction...	Risk Increase...
LOSP	2	LOSP	11	2.300E+000	1.000E+000	1.900E+038	4.348E-001
		DG-A	1	2.000E-002	9.494E-001	1.975E+001	4.752E+001
		E-MOV-1	1	1.000E-003	4.657E-002	1.049E+000	4.752E+001
		E-MOV-A	3	5.000E-003	1.884E-003	1.002E+000	1.374E+000
		E-MOV-B	3	5.000E-003	1.884E-003	1.002E+000	1.374E+000
		E-PUMP-A	3	3.000E-003	1.130E-003	1.001E+000	1.375E+000
		E-PUMP-B	3	3.000E-003	1.130E-003	1.001E+000	1.375E+000
		E-CV-A	3	1.000E-004	3.768E-005	1.000E+000	1.376E+000
		E-CV-B	3	1.000E-004	3.768E-005	1.000E+000	1.376E+000

16 BASE CASE UPDATE

Section 16 describes how to perform a base case update on event tree accident sequence results and fault tree results.

16.1. Base Case Update

All results generated by SAPHIRE are stored in the current case. These results change **each** time a new analysis is performed. To save the results (permanently) that were just generated, a base case update needs to be performed.

Base case update is a means to store data and results in the family files as a "permanent" record.

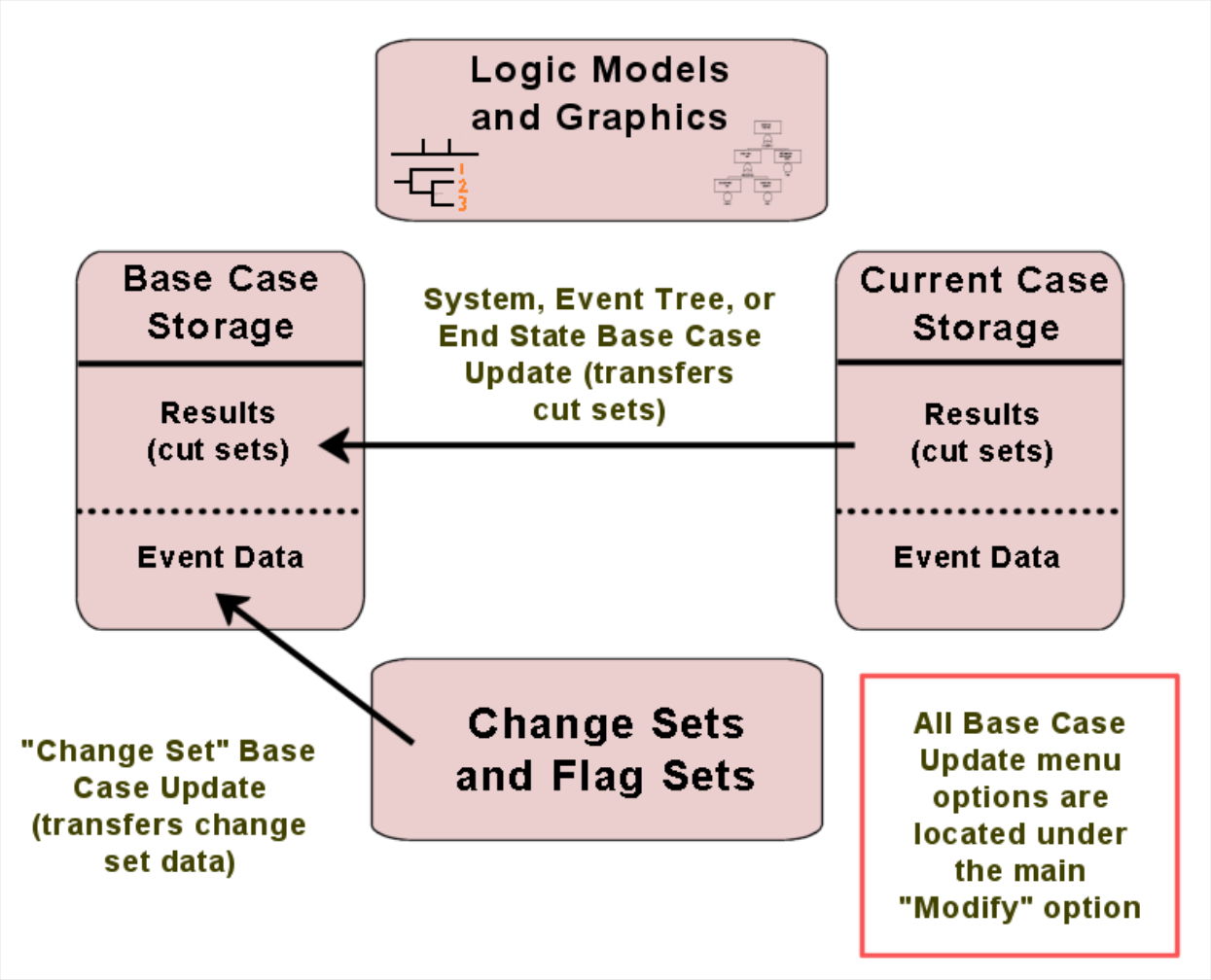
Base case data and results are updated by the transferring of current case results to the base case. Transferring the current case data and results into the base case gives a "permanent" record that can not be removed until another base case update is performed.

By performing a base case update, these base case results can be readily compared to results from other sensitivity analyses.

A base case update option is available for:

- Fault Trees
- Event Trees (sequences)
- End States
- Change Sets

The various base case updates are shown graphically below.



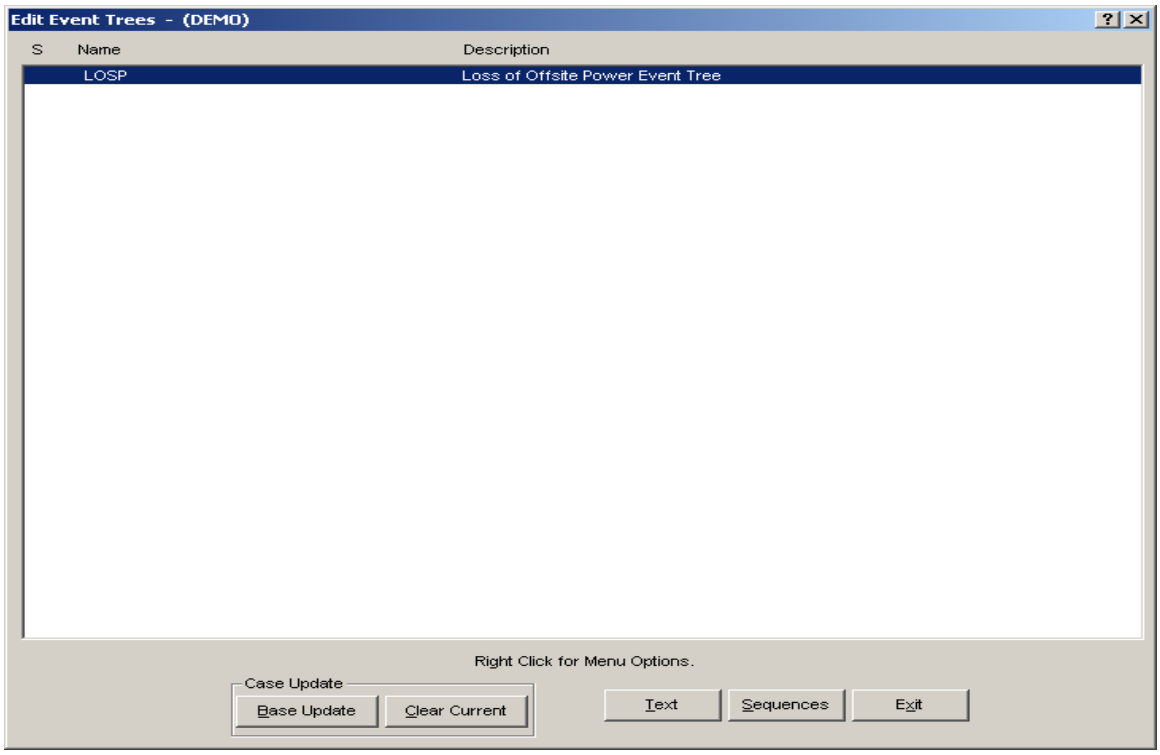
16.2. Base case update for accident sequences

The prerequisites for performing a sequence base case update are:

1. Generate appropriate data changes for the analysis.
2. Analyze (i.e., generate cut sets) the accident sequences at a specified truncation level.
3. Apply recovery rules to the accident sequence cut sets if required.
4. Review resulting cut sets for correctness.
5. Perform uncertainty analysis if needed.

We are now ready to perform a base case update.

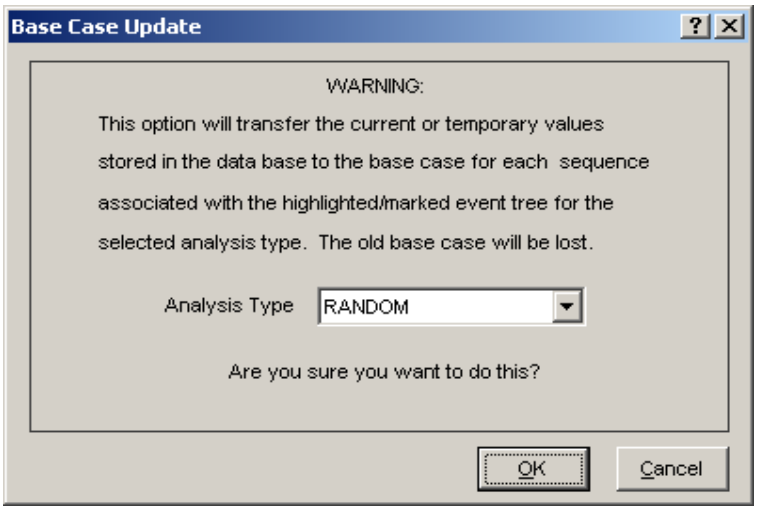
Select the **Modify ➔ Event Trees** menu.



Mark the event trees to be updated, then choose the **Base Update** button.

(The **Clear Current** button removes the results stored in the current case.)

Base Case Update Dialog



Choose the **OK** button to perform the base case update.

The Analysis Type drop-down list gives you the opportunity to do a base case update on any one of the sixteen different analysis types.

Note: The default analysis type as shown in the "Define Constants" (see Section 1) will be the default selection for the base case updated.

To perform a base case update on any of the other analysis types, you must select the analysis type prior to choosing the **OK** button.

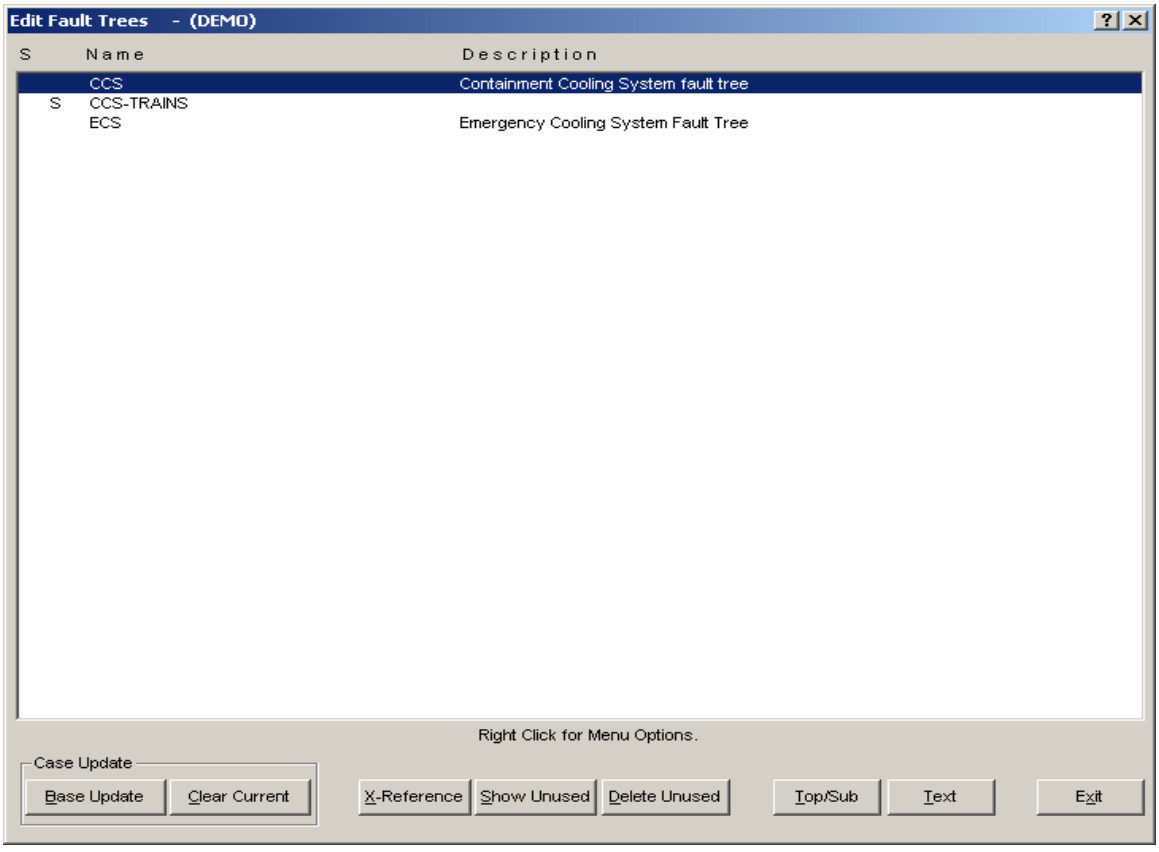
16.3. Base case update for fault tree results

The prerequisites for performing a fault tree base case update are:

1. Generate appropriate data changes for the analysis.
2. Analyze (i.e., generate cut sets) the fault trees at a specified truncation level.
3. Apply recovery rules to the fault tree cut sets if required.
4. Review resulting cut sets for correctness.
5. Perform uncertainty analysis if needed.

We are now ready to perform a base case update.

Select the **Modify ➔ Fault Trees** menu.



Mark the fault trees, then choose the **Base Update** button.

(The **Clear Current** button removes the results stored in the current case.)

17 Fault Tree Sensitivity Analysis

Section 17 describes how to perform fault tree (system) sensitivity studies including basic event data modifications and fault tree logic changes. The use of Change Sets to make basic event data modifications is described.

17.1. Overview of Steps Involved in Performing a Fault Tree Sensitivity Analysis

If fault tree logic changes are to be made (e.g., adding a basic event, removing a basic event, or changing an OR-gate to an AND-gate), make the changes using the graphical fault tree editor or fault tree logic editor.

If data changes are to be made, enter data modifications by either

- ◊ Changing the data "permanently" in the **Modify ➔ Basic Events** option
- ◊ Changing the data "temporarily" using Change Sets

Then, use the **Generate** option to "process" the basic event changes so that they will be used for subsequently performed SAPHIRE operations such as quantifying cut sets, displaying cut sets, and generating reports.

Update the fault tree cut sets in the **Fault Tree** menu using the **Solve** option.

Modifying Fault Tree Logic

The fault tree logic can be changed by using the fault tree graphical editor by selecting the **Fault Tree ➔ Edit Graphics** option.

Alternate ways of modifying fault tree logic include:

- ◊ Using the **Fault Tree ➔ Edit Logic** option also discussed in Section 5.
- ◊ Replacing system logic with MAR-D .FTL files.

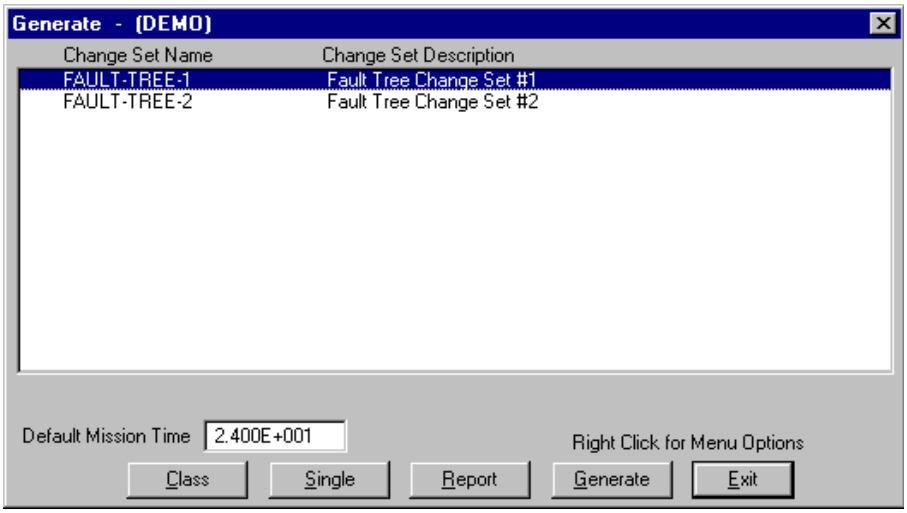
17.2. Making a "Permanent" Data Change in MODIFY Data Base

- To enter basic event data, select **Modify** from the menu bar. Then select **Basic Events**.
  - ◊ To modify data for an existing event, highlight the event you want to edit, right-click to invoke the pop-up menu, and select the **Modify** option.
  - ◊ The **Generate** option from the main menu bar must be used before the data is available for SAPHIRE operations.

  It is not recommended that permanent changes be used for sensitivity analyses.

17.3. Making a "Temporary" Data Change by Using Change Sets

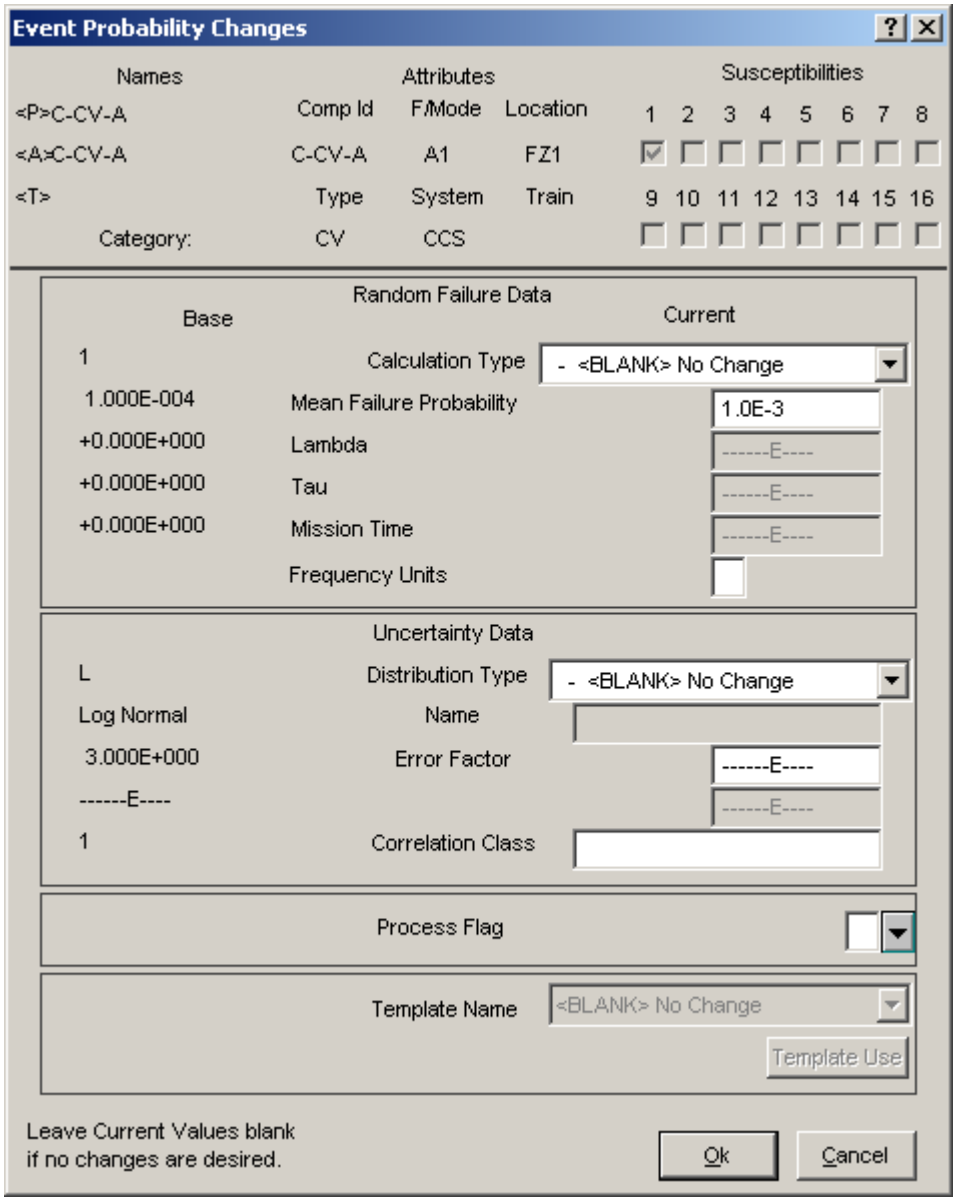
- Select **Generate** from the menu bar.
- To create a Change Set, right-click to invoke the pop-up menu, select **Add**, and enter the Change Set name and description.
- As shown in the figure, two Change Sets (named FAULT-TREE-1 and FAULT-TREE-2) were created in this example.



- To make a data change in the Change Set named FAULT-TREE-1, highlight the Change Set, choose the **Single** button.

- Right-click to invoke the pop-up menu and select the **Add** option. Double-click on the event to be modified. In this example, event C-CV-A was edited.

The Change Set Data Entry Dialog

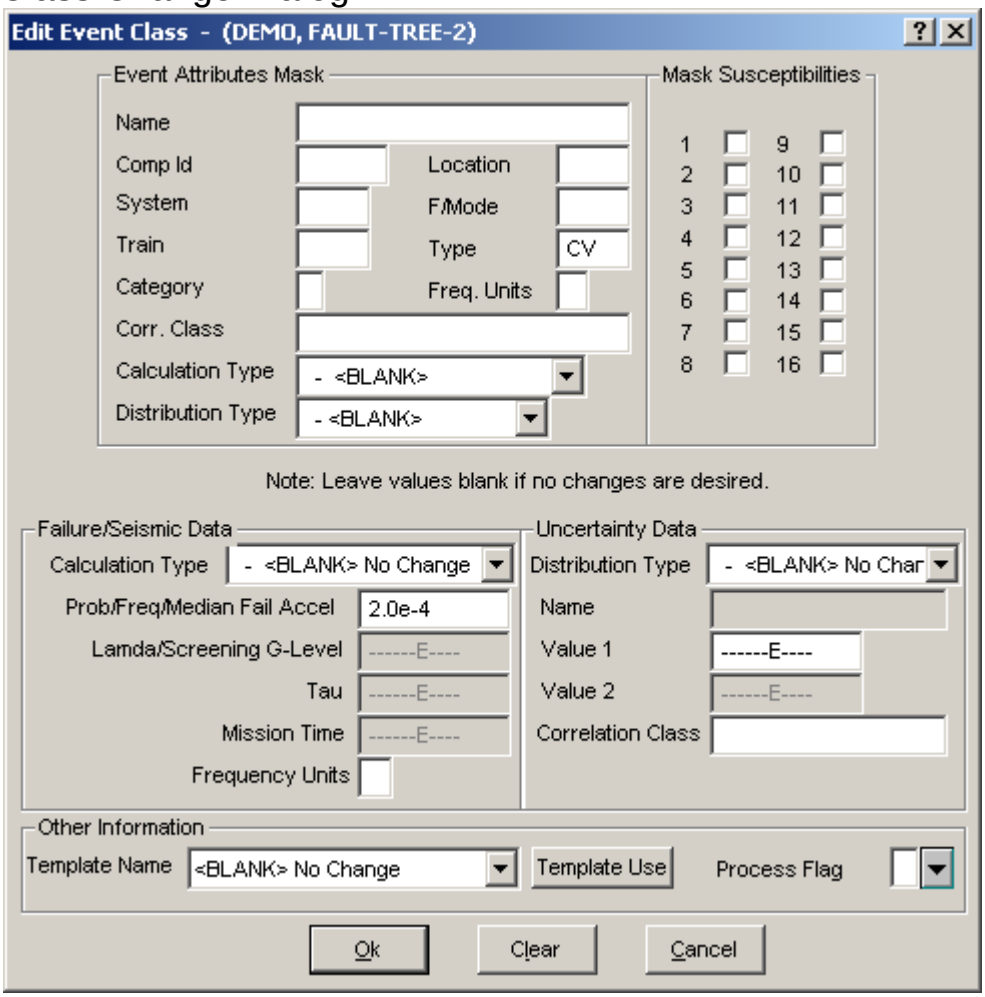


- The Event Probability Changes dialog allows you to enter changes on the right side of the dialog.

- In this example, the probability field was changed from 1.0E-4 to 1.0E-3.

- To create a Class Change for the FAULT-TREE-2 Change Set, highlight the Change Set, choose the **Class** button.

The Class Change Dialog

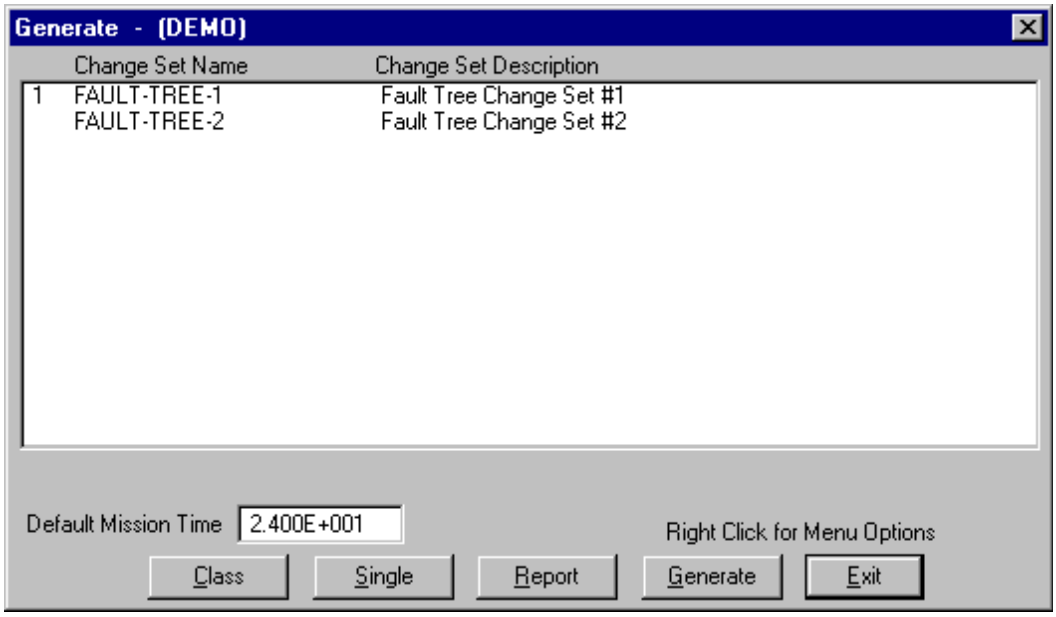


- Enter the attributes that define the events to be changed. In this example, Type was specified as CV.

- Enter the new data. In this example, the probability was changed to 2.0E-4.

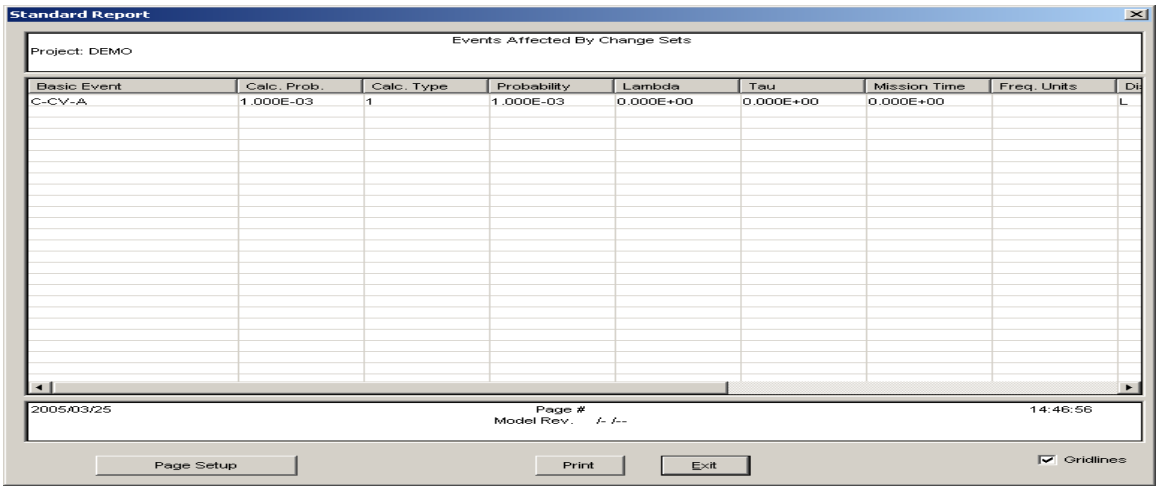
- One class change is allowed for each Change Set.

Generating Changes for One Change Set

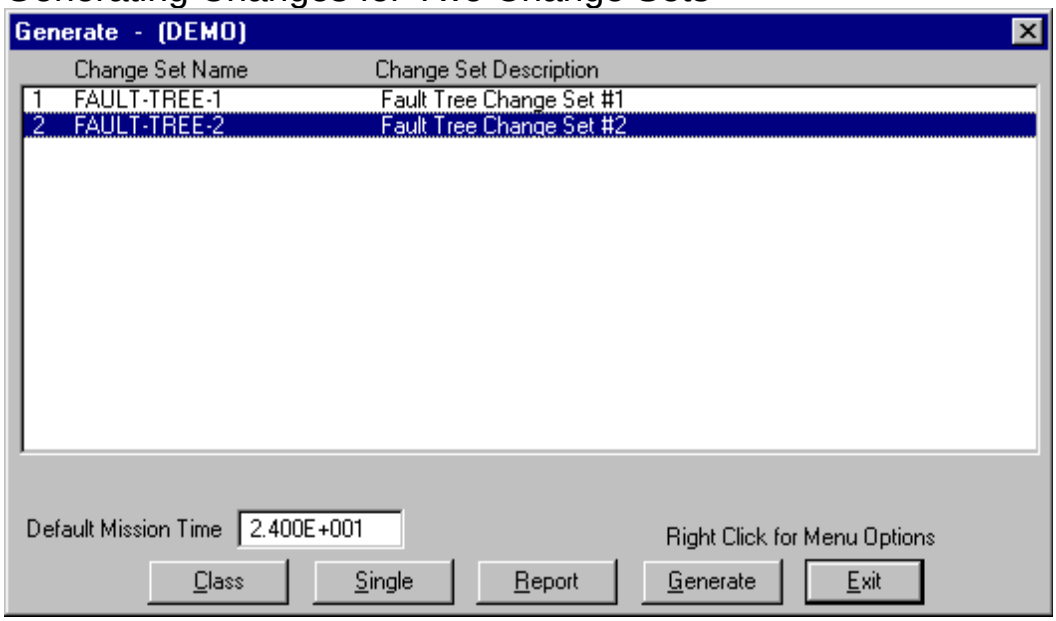


- To invoke only the FAULT-TREE-1 Change Set, double-click on the Change Set.
- The number 1 will appear in the left column.
- Choose the **Generate** button and proceed as directed.
- To report the changes, choose the **Report** button, then select the Affected Events radio button.

Report Changes

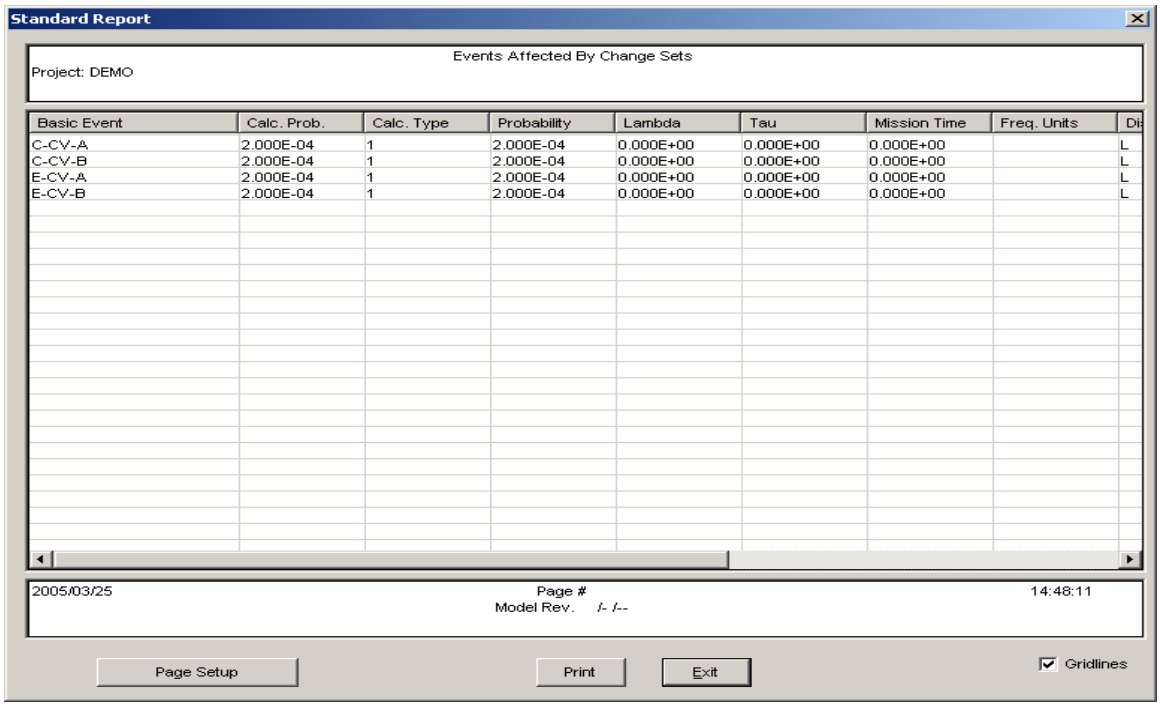


Generating Changes for Two Change Sets

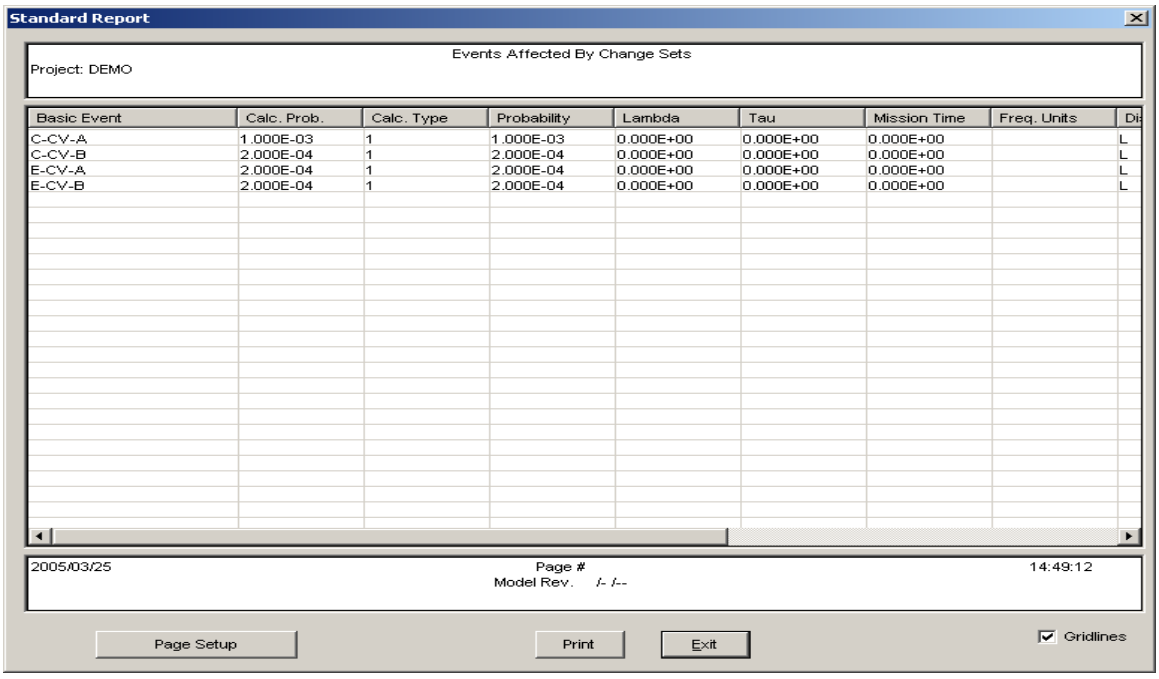


- When invoking more than one Change Set, the order that they are selected dictates which changes have precedence.
- In this example, the FAULT-TREE-1 Change Set was marked, and then the FAULT-TREE-2 Change Set was marked.
- Note the numbering in the left column. This will cause the FAULT-TREE-2 changes to override the FAULT-TREE-1 changes where the same basic event is affected by both Change Sets.
- Choose the **Generate** button and continue as directed.
- To report the changes, choose the **Report** button, then select the Affected Events radio button.

Generating Changes - Compare Reports

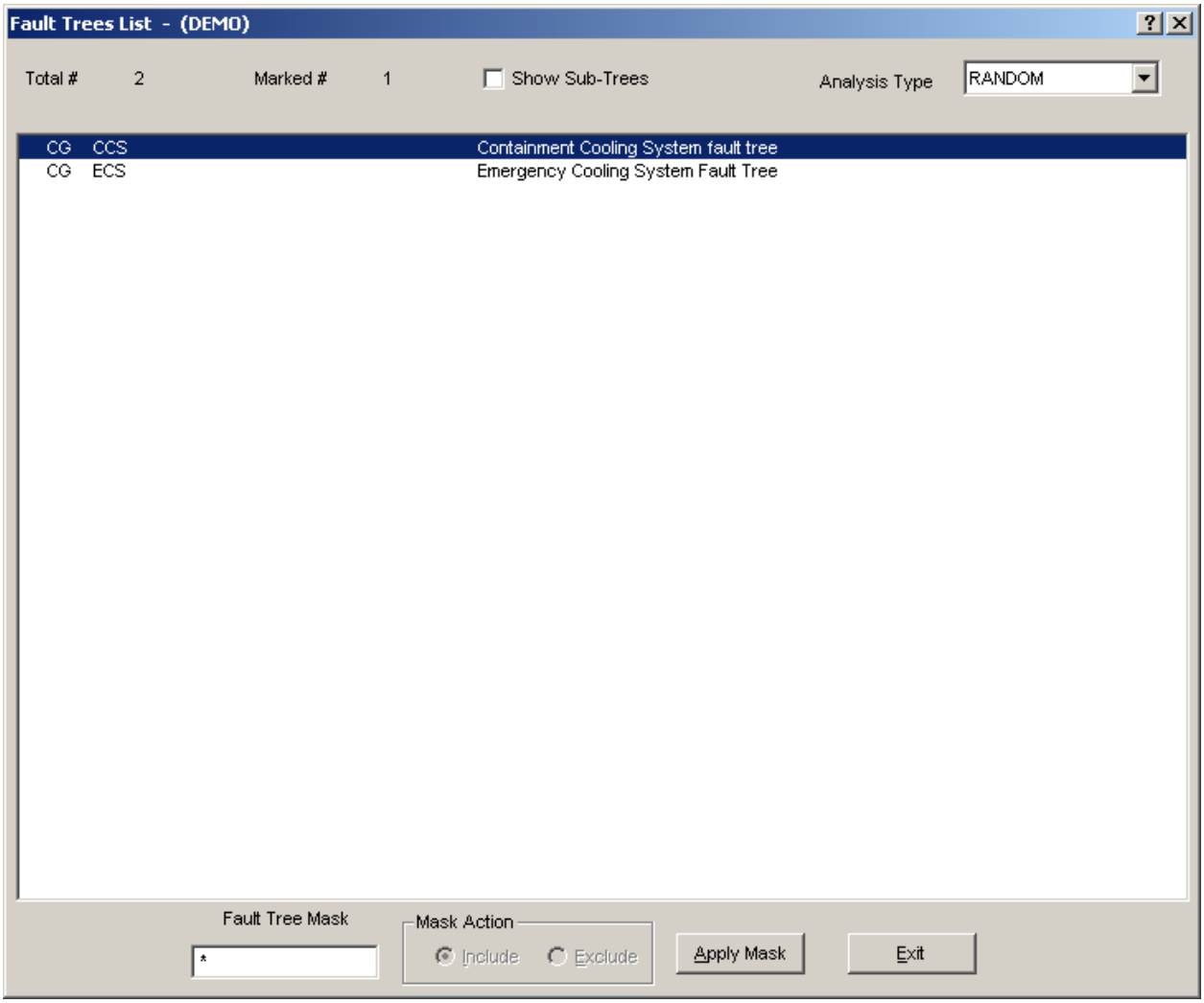


In the example above, the FAULT-TREE-1 had been selected first, and then FAULT-TREE-2.



If the FAULT-TREE-2 had been selected first, and then FAULT-TREE-1, the resulting affected basic events would be as shown above.

17.4. Analyzing Fault Tree Cut Sets



- Select **Fault Tree** from the menu bar.
- Mark the fault trees using the mask function, or individually using the mouse.
- Right-click to invoke the pop-up menu.
- Select the **Solve** option from the pop-up menu.

18 Event Tree Sensitivity Analysis

Section 18 describes how to perform event tree sensitivity studies including basic event data modifications and event tree logic changes. The use of Change Sets to make basic event data modifications is described.

18.1. Overview of Steps Involved in Performing an Event Tree Sensitivity Analysis

If event tree logic changes are to be made, make changes to the event tree logic (e.g., add or delete a top event, or modify the event tree branches), and then use **Event Tree ➔ Link Trees** to link the new event tree logic.

If fault tree logic changes are to be made, make changes to the fault tree logic using the graphical fault tree or logic editor.

If data changes are to be made, enter data modifications by either

- ◊ Changing the data "permanently" in the **Modify ➔ Basic Events** menu,
- ◊ Changing the data "temporarily" using Change Sets.

Then, use the **Generate** option from the menu bar to "process" the basic event changes so that they will be used for subsequently performed SAPHIRE operations such as quantifying and displaying cut sets and generating reports.

To update the event tree sequence cut sets select **Sequence** from the menu bar. Then right-click to invoke the pop-up menu and select the **Solve** option.

18.2. Modifying Event Tree Logic (or Fault Tree Logic Affecting the Event Tree)

Event tree logic can be changed by using the graphical editor in the **Event Tree ➔ Edit Graphics** menu.

Fault tree logic can be changed by using the graphical editor in the **Fault Tree ➔ Edit Graphics**.

Alternate ways of modifying event tree logic include:

- ◊ Using the **Event Tree ➔ Edit Logic**
- ◊ Replacing sequence logic with MAR-D .SQL files

18.3. Making a "Permanent" Data Change in MODIFY Data Base

To enter basic event data, select **Modify** from the menu bar. Then select **Basic Events**.

- ◊ To modify data for an existing event, highlight the event you want to edit, right-click to invoke the pop-up menu, and select the **Modify** option.

- ◊ The **Generate** option from the main menu bar must be used before the data is available for SAPHIRE operations.

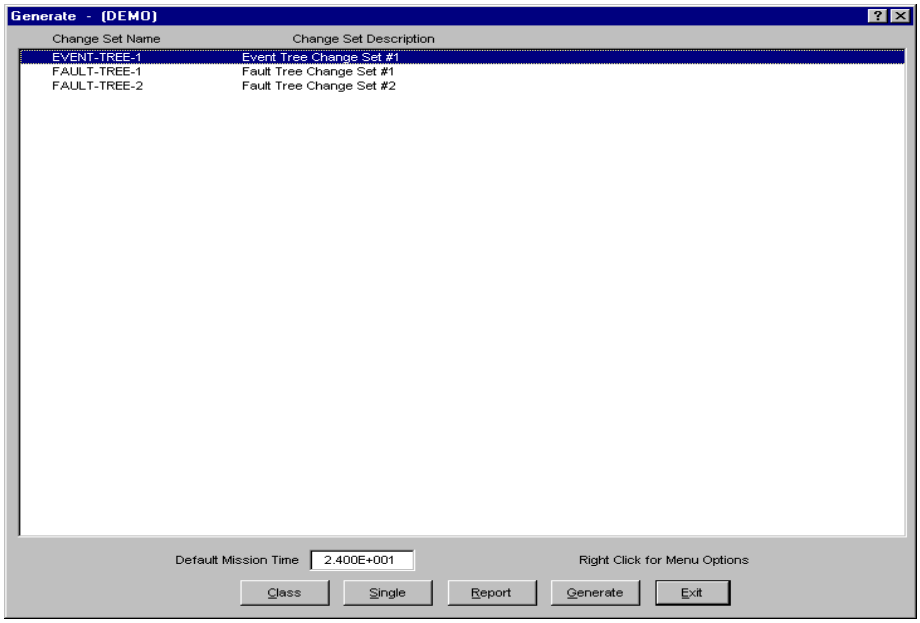
Note: It is not recommended that permanent changes be used for sensitivity analyses.

18.4. Making a "Temporary" Data Change by Using Change Sets

Select **Generate** from the menu bar.

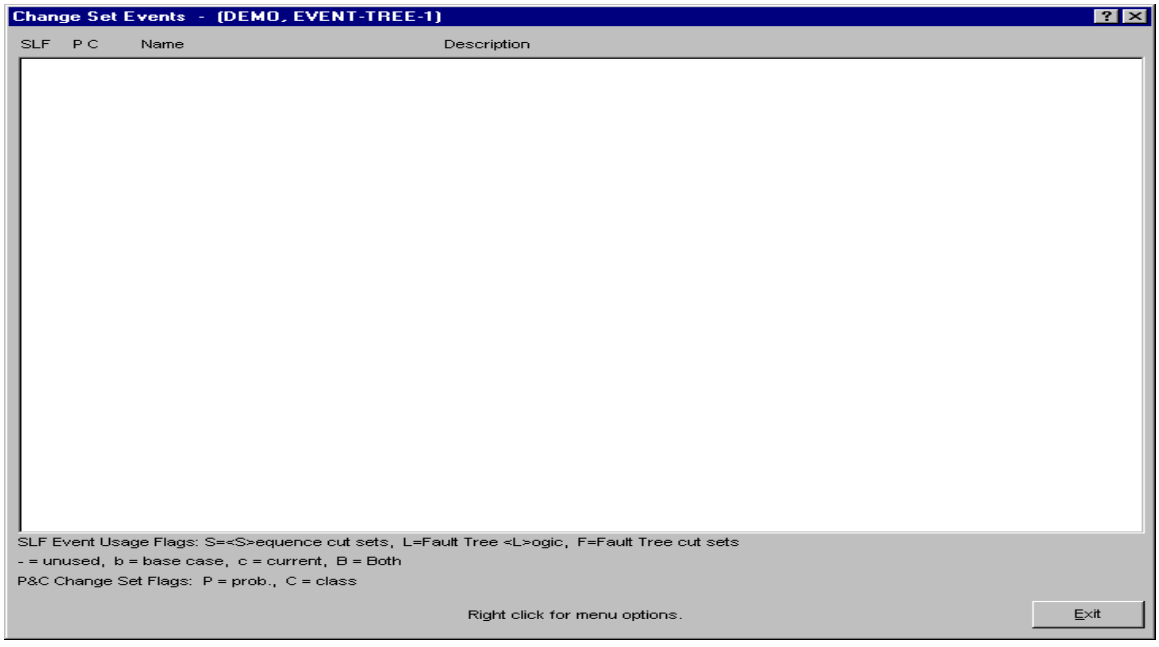
To create a Change Set, right-click to invoke the pop-up menu, select **Add**, and enter the Change Set name and description.

Three Change Sets are shown in the figure (named FAULT-TREE-1, FAULT-TREE-2 and EVENT-TREE-1). EVENT-TREE-1 will be created for this example.



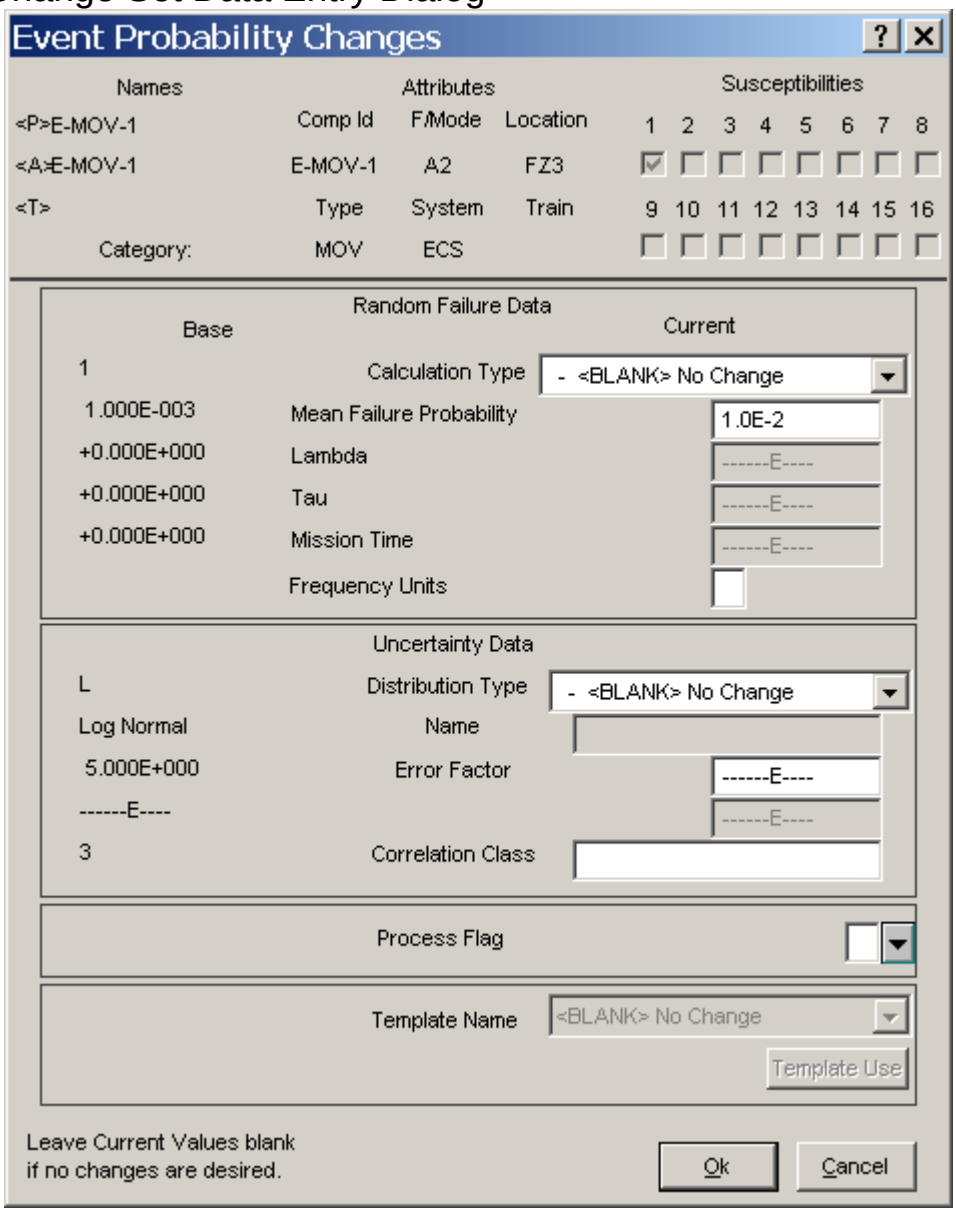
To make a data change in the Change Set named EVENT-TREE-1, highlight the Change Set, choose the **Single** button.

The Change Set Select Event Dialog



Right-click to invoke the pop-up menu and select the Add option. Double-click on the event to be modified. In this example, event E-MOV-1 was edited.

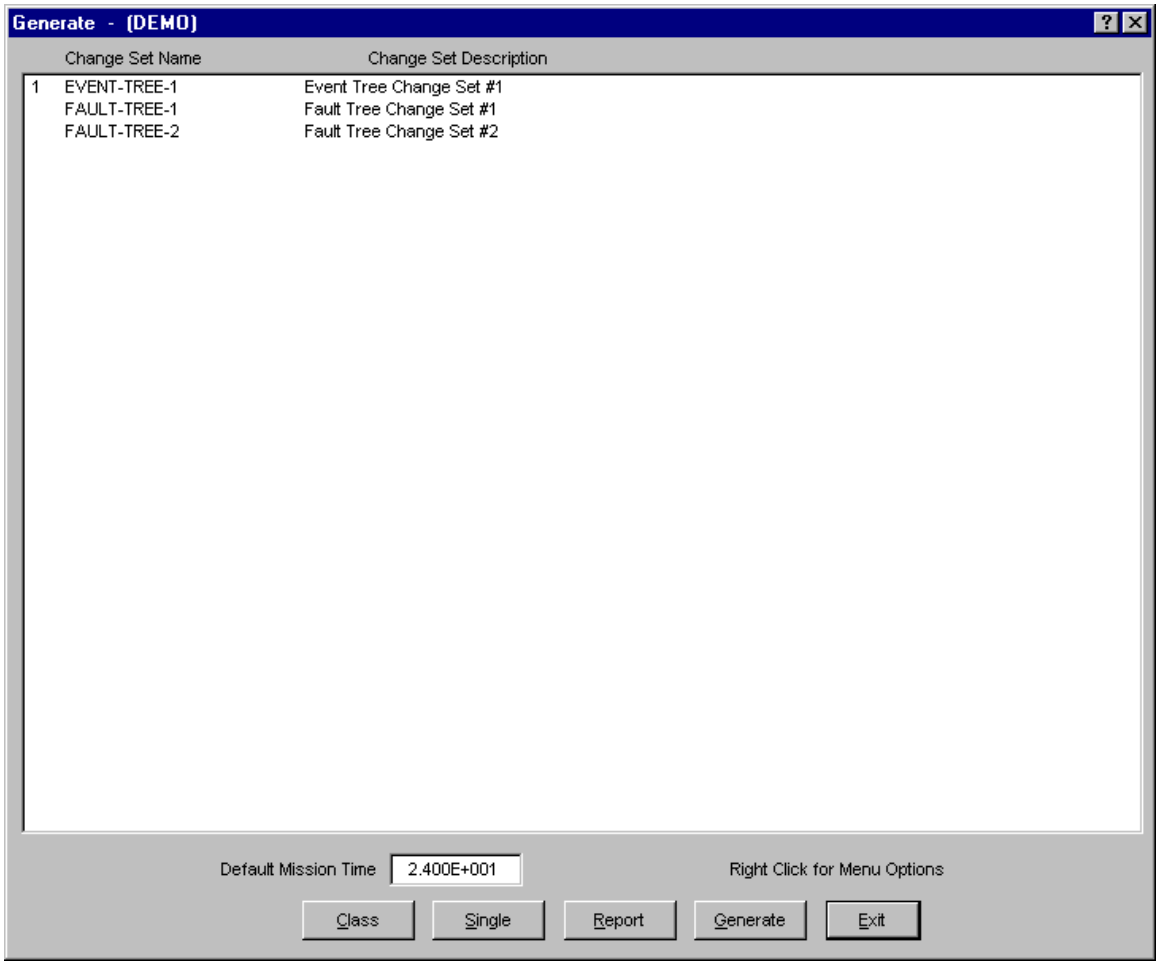
The Change Set Data Entry Dialog



The Event Probability Changes dialog allows you to enter changes on the right side of the dialog.

In this example, the probability field was changed from 1.0E-3 to 1.0E-2.

Generating Changes for One Change Set



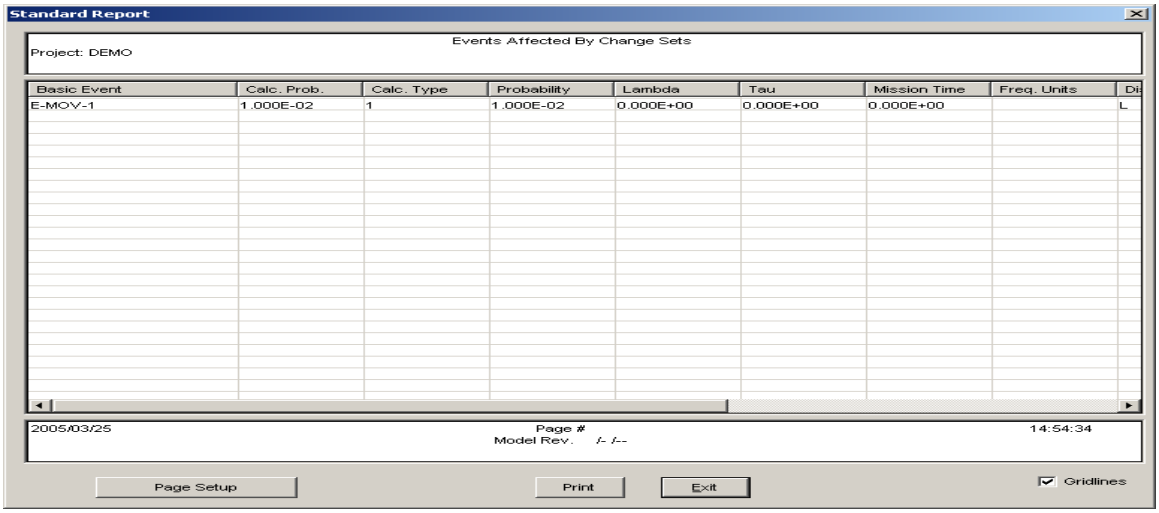
To invoke only the EVENT-TREE-1 Change Set, double-click on the Change Set.

The number 1 will appear in the left column.

Choose the **Generate** button and proceed as directed.

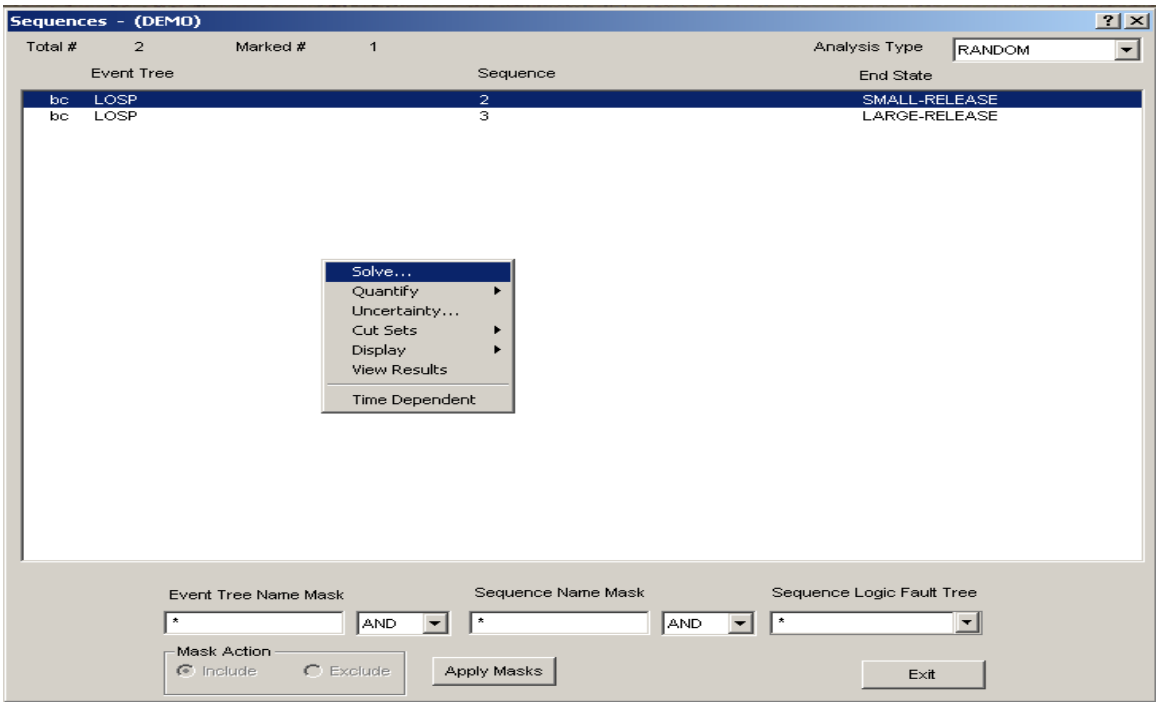
To report the changes, choose the **Report** button, then select the Affected Events radio button.

Report Changes



18.5. Analyzing Event Tree Sequence Cut Sets

Select **Sequence** from the menu bar.



Mark the sequences using the mask features, or individually using the mouse.

Right-click to invoke the pop-up menu and select the **Solve** option. The minimal cut sets will be generated via this option.

19 USING DATABASE FILES

Section 19 describes ways to transfer event tree, fault tree, or basic event information from one project database to either a newly created project database or an existing project database.

19.1. Event Tree MAR-D Files

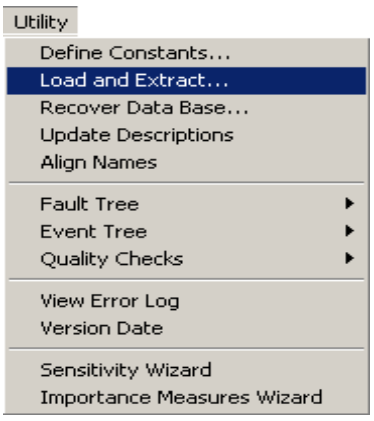
The event tree files created in SAPHIRE have the suffix of *.ET?, where the ? indicates the "sub" file type. For example, the file labeled .ETD contains event tree **D**escriptions.

These MAR-D files store information about the event trees including logic, rules, and attributes.

- The *.ETG file is created when the event tree is built and the logic is saved.
- All other *.ET* files are created only by using the MAR-D Extract feature.
- SAPHIRE allows information pertaining to the event trees to be extracted from one project and copied to another SAPHIRE project.
- The MAR-D module in SAPHIRE allows the user to extract event tree information to be edited using a text editor.
- MAR-D also allows the user to load extracted files from another project into the current project.

19.2. MAR-D Load and Extract Menus

The MAR-D menus are provided in the **Utility ➔ Load and Extract** menu.

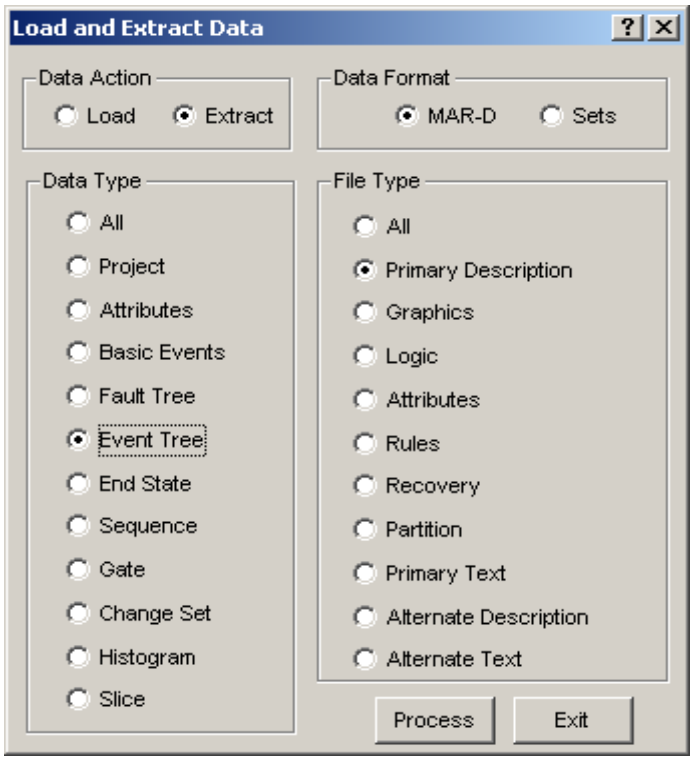


Extract Data
This option allows you to extract MAR-D files from the database into a text file.

Load Data

This option allows you to load MAR-D files that are contained in the project's subdirectory.

Load and Extract Event Tree Data



From the Data **Action** section, choose either the Load or Extract radio button.

From the Data **Format** section, choose either the MAR-D or Sets radio button. When dealing with SAPHIRE, MAR-D is the preferred format.

Then choose the desired data type from the Data **Type** section.

The event tree information MAR-D format is selected in the figure shown above. Any one of the listed MAR-D file types can be accessed by selecting the appropriate radio button.

- Choose the **Process** button.
- You can mark individual event trees, a range of event trees, or all of the event trees.
- Choose the **Extract/Load** button

- When extracting, you will be prompted to accept or change the MAR-D file name. The default name is usually the project name and the MAR-D file 3-character extension (e.g., DEMO.ETD).
- When loading, the files in the project directory that have the extension *.ET? will be listed on the dialog. Mark the file (or files) to load.

19.3. Fault Tree MAR-D Files

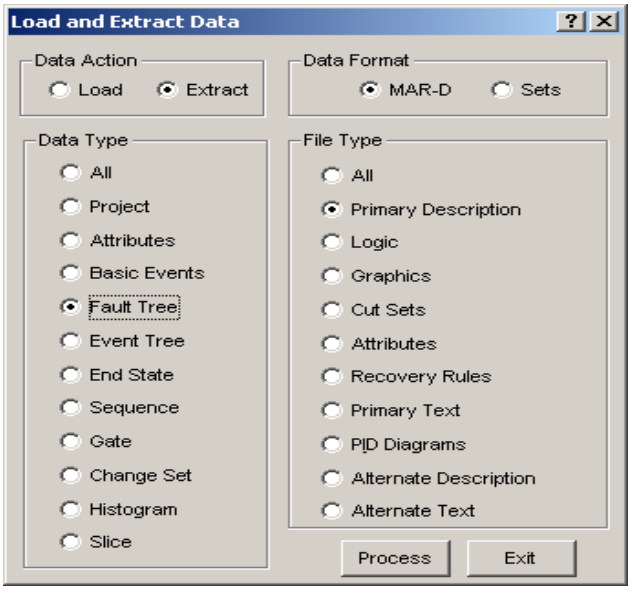
The fault tree files created in SAPHIRE have the suffix of *.FT? and *.DLS. The .DLS is a binary file used to store the fault tree graphic.

The files store information about the fault trees, including logic, rules, and attributes.

- The *.DLS files are created when the fault tree is built and the logic is saved.
- All other *.FT? files are created only using the MAR-D Extract feature.
- SAPHIRE allows information pertaining to the fault trees to be extracted from one project and copied to another SAPHIRE project.
- The MAR-D module in SAPHIRE allows the user to extract fault tree information to be edited using a text editor.
- MAR-D also allows the user to load extracted files from another project into the current project.

Load and Extract Fault Tree Data

From the Data **Action** section, choose either the Load or Extract radio button.
From the Data **Format** section, choose either the MAR-D or Sets radio button.
Then choose the desired data type from the Data **Type** section.



- ◊ Choose the **Process** button.
- ◊ You can mark individual fault trees, a range of fault trees, or all of the fault trees.
- ◊ Choose the **Extract/Load** button.
- ◊ When extracting, you will be prompted to accept or change the MAR-D file name. The default name is usually the project name and the MAR-D file 3-character extension (e.g., DEMO.FTD).
- ◊ When loading, the files in the project directory that have the extension *.FT? and *.DLS will be listed on the dialog. Mark the file (or files) to load.

19.4. Basic Event MAR-D Files

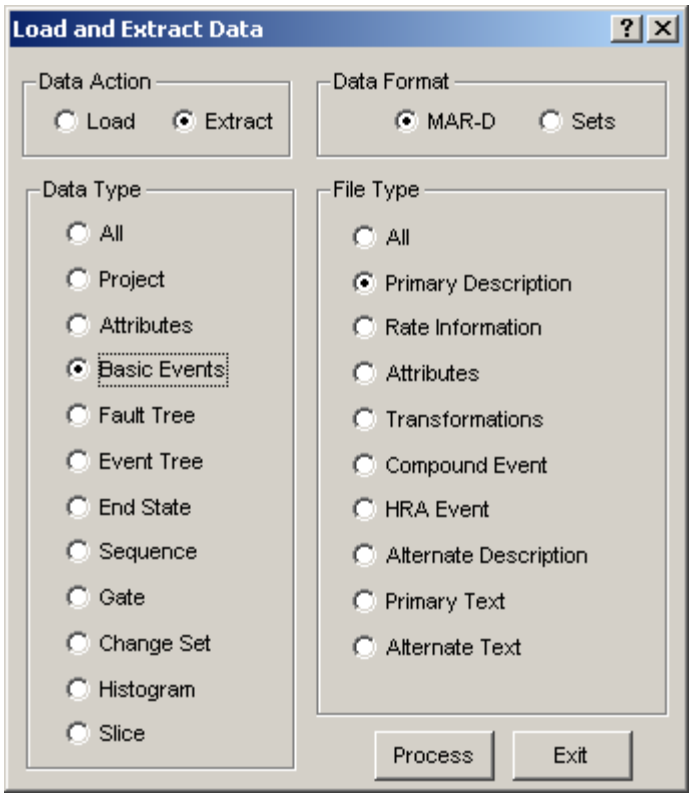
The basic event files created in SAPHIRE have the suffix of *.BE?.

These files store information about the basic events, including probability information, attributes, and descriptions.

- ♦ The *.BE* files are created only by using the MAR-D Extract feature.
- ♦ SAPHIRE allows information pertaining to the basic events to be extracted from one project and copied to another SAPHIRE project.
- ♦ The MAR-D module in SAPHIRE allows the user to extract basic event information to be edited using a text editor.
- ♦ MAR-D also allows the user to load extracted files from another project to the current project.

Load and Extract Basic Event Data

From the Data **Action** section, choose either the Load or Extract radio button.
From the Data **Format** section, choose either the MAR-D or Sets radio button.
Then choose the desired data type from the Data Type section.



- ◊ Choose the **Process** button.
- ◊ You can mark individual basic events, a range of basic events, or all of the basic events.
- ◊ Choose the **Extract/Load** button.
- ◊ When extracting, you will be prompted to accept or change the MAR-D file name. The default name is usually the project name and the MAR-D file 3-character extension (e.g., DEMO.BED).
- ◊ When loading, the files in the project directory that have the extension *.BE? will be listed on the dialog. Mark the file (or files) to load.

19.5. Moving database files between projects

Database information from one SAPHIRE project can be moved into another SAPHIRE project via MAR-D. The steps involved in this transfer are:

1. Perform the MAR-D output from the “primary” project (the one to copy *from*).
2. Copying (in Windows, see the note below) the extracted MAR-D files to another SAPHIRE project subdirectory.

3. Return to SAPHIRE and open the second project (the one that has the copy of the MAR-D files).
4. Select the **Utility ➔ Load and Extract** menu.
5. Select the "Load" option and the applicable information to be loaded.
6. Choose the **Process** button, then indicate the MAR-D file to be loaded. Following this step, the data will now appear in this second project.
7. Recover the database from the **Utility ➔ Recover Data Base** menu, just to ensure that the relational database files are intact.
8. Generate event data from the **Generate** menu.

Note: To copy files, start Windows explorer or browse "My Computer." Find the subdirectory containing the MAR-D files. Highlight all of the files by holding down the <Ctrl> key and clicking on each file to be copied. Press CTRL-C or "Edit ➔ Copy" from the menu options in Windows Explorer. Move to the subdirectory of the second project, and press CTRL-V or "Edit ➔ Paste" to paste the files into the directory.