Systems Analysis Programs for Hands-On Integrated Reliability Evaluations (SAPHIRE) Summary Manual

S. T. Wood
C. L. Smith
K. J. Kvarfordt
S. T. Beck

August 2008

INL

Idaho National Laboratory

Systems Analysis Programs for Hands-On Integrated Reliability Evaluations (SAPHIRE) Summary Manual

S. T. Wood
C. L. Smith
K. J. Kvarfordt
S. T. Beck

August 2008

Idaho National Laboratory
Idaho Falls, Idaho 83415

http://www.inl.gov

AVAILABILITY NOTICE

Availability of Reference Materials Cited in NRC Publications

Most documents cited in NRC publications will be available from one of the following sources:

1. The NRC Public Document Room, 11555 Rockville Pike, Rockville, MD 20852 (pdr@nrc.gov)

2. The Superintendent of Documents, U. S. Government Printing Office (GPO), Mail Stop SSOP, Washington, DC 20402-9328

3. The National Technical Information Service, Springfield, VA 22161

Although the listing that follows represents the majority of documents cited in NRC publications, it is not intended to be exhaustive.

Referenced documents available for inspection and copying for a fee from the NRC Public Document Room include NRC correspondence and internal NRC memoranda; NRC bulletins, circulars, information notices, inspection and investigative notices; licensee event reports; vendor reports and correspondence; Commission papers; and applicant and licensee documents and correspondence.

The following documents in the NUREG series are available for purchase from the GPO Sales Program: formal NRC staff and contractor reports, NRC-sponsored conference proceedings, international agreement reports, grant publications, and NRC booklets and brochures. Also available are regulatory guides, NRC regulations in the *Code of Federal Regulations*, *and Nuclear Regulatory Commission Issuances*.

Documents available from the National Technical Information Service include NUREG-series reports and technical reports prepared by other Federal agencies and reports prepared by the Atomic Energy Commission, forerunner agency to the Nuclear Regulatory Commission.

Documents available from public and special technical libraries include all open literature items, such as books, journal articles, and transactions. *Federal Register* notices, Federal and State legislation, and congressional reports can usually be obtained from these libraries.

Documents such as theses, dissertations, foreign reports and translations, and non-NRC conference proceedings are available for purchase from the organization sponsoring the publication cited.

Single copies of NRC draft reports are available free, to the extent of supply, upon written request to the Office of Administration, Distribution and Mail Services Section U. S. Nuclear Regulatory Commission, Washington, DC 20555-0001.

The public maintains copies of industry codes and standards used in a substantive manner in the NRC regulatory process at the NRC Library, Two White Flint North, 11545 Rockville Pike, Rockville, MD, 20852, for use. Codes and standards are usually copyrighted and may be purchased from the originating organization or, if they are American National Standards, from the American National Standards Institute, 1430 Broadway, New York, NY 10018.

DISCLAIMER NOTICE

PREVIOUS REPORTS

Smith, C. L., et al., *Testing, Verifying, and Validating SAPHIRE Versions 6.0 and 7.0*, NUREG/CR-6688, October 2000.

K. D. Russell, et al. *Systems Analysis Programs for Hands-on Reliability Evaluations (SAPHIRE) Version 6.0 - System Overview Manual*, NUREG/CR-6532, May 1999.

K. D. Russell et al., *Integrated Reliability and Risk Analysis System (IRRAS) Version 5.0, Volume 2 - Reference Manual*, NUREG/CR-6116, EGG-2716, July 1994.

K. D. Russell et al., *Verification and Validation (V&V), Volume 9 – Reference Manual*, NUREG/CR-6116, EGG-2716, July 1994.

K. D. Russell et al., *Integrated Reliability and Risk Analysis System (IRRAS) Version 4.0, Volume 1 - Reference Manual*, NUREG/CR-5813, EGG-2664, January 1992.

K. D. Russell et al., *Integrated Reliability and Risk Analysis System (IRRAS) Version 2.5 Reference Manual*, NUREG/CR-5300, EGG-2613, March 1991.

K. D. Russell, M. B. Sattison, D. M. Rasmuson, *Integrated Reliability and Risk Analysis System (IRRAS) - Version 2.0 User's Guide*, NUREG/CR-5111, EGG-2535, manuscript completed March 1989, published June 1990.

K. D. Russell, D. M. Snider, M. B. Sattison, H. D. Stewart, S.D. Matthews, K. L. Wagner, *Integrated Reliability and Risk Analysis System (IRRAS) User's Guide - Version 1.0 (DRAFT)*, NUREG/CR-4844, EGG-2495, June 1987.

ABSTRACT

The Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE) is a software application developed for performing a complete probabilistic risk assessment (PRA) using a personal computer (PC) running the Microsoft Windows operating system. SAPHIRE is primarily funded by the U.S. Nuclear Regulatory Commission (NRC) and developed by the Idaho National Laboratory (INL). INL's primary role in this project is that of software developer and tester. However, INL also plays an important role in technology transfer by interfacing and supporting SAPHIRE users, who constitute a wide range of PRA practitioners from the NRC, national laboratories, the private sector, and foreign countries.

SAPHIRE can be used to model a complex system's response to initiating events and quantify associated consequential outcome frequencies. Specifically, for nuclear power plant applications, SAPHIRE can identify important contributors to core damage (Level 1 PRA) and containment failure during a severe accident which lead to releases (Level 2 PRA). It can be used for a PRA where the reactor is at full power, low power, or at shutdown conditions. Furthermore, it can be used to analyze both internal and external initiating events and has special features for transforming an internal events model to a model for external events, such as flooding and fire analysis. It can also be used in a limited manner to quantify risk in terms of release consequences to the public and environment (Level 3 PRA).

SAPHIRE also includes a separate module called the Graphical Evaluation Module (GEM). GEM is a special user interface linked to SAPHIRE that automates the SAPHIRE process steps for evaluating operational events at commercial nuclear power plants. Using GEM, an analyst can estimate the risk associated with operational events (for example, to calculate a conditional core damage probability) very efficiently and expeditiously.

This report provides an overview of the functions available in SAPHIRE and presents general instructions for using the software. Section 1 presents SAPHIRE's historical evolution and summarizes its capabilities. Section 2 presents instructions for installing and using the code. Section 3 explains the database structure used in SAPHIRE and discusses database concepts. Section 4 explains how PRA data (event frequencies, human error probabilities, etc.) can be generated and manipulated using "change sets." Section 5 deals with fault tree operations, including constructing, editing, solving, and displaying results. Section 6 presents operations associated with event trees, including rule application for event tree linking, partitioning, and editing sequences. Section 7 presents how accident sequences are generated, solved, quantified, and analyzed. Section 8 discusses the functions available for performing end state analysis. Section 9 explains how to modify data stored in a SAPHIRE database. Section 10 illustrates how to generate and customize reports. Section 11 covers SAPHIRE utility options to perform routine functions such as defining constant values, recovering databases, and loading data from external sources. Section 12 provides an overview of GEM's features and capabilities. Finally, Section 13 summarizes SAPHIRE's quality assurance process.

FOREWORD

The U.S. Nuclear Regulatory Commission has developed the Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE) software used to perform probabilistic risk assessments (PRAs) on a personal computer. SAPHIRE enables users to supply basic event data, create and solve fault and event trees, perform uncertainty analyses, and generate reports. In that way, analysts can perform PRAs for any complex system, facility, or process.

SAPHIRE can be used to model a plant's response to initiating events, quantify core damage frequencies, and identify important contributors to core damage (Level 1 PRA). The program can also be used to evaluate containment failure and release models for severe accident conditions, given that core damage has occurred (Level 2 PRA). In so doing, the analyst could build the PRA model assuming that the reactor is initially at full power, low power, or shutdown. In addition, SAPHIRE can be used to analyze both internal and external events, and it includes special features for transforming models built for internal event analysis to models for external event analysis. It can also be used in a limited manner to quantify the frequency of release consequences (Level 3 PRA). Because this software is a very detailed technical tool, users should be familiar with PRA concepts and methods used to perform such analyses.

SAPHIRE has evolved with advances in computer technology. The versions currently in use (6 and 7) run in the Microsoft Windows® environment. A user-friendly interface, Graphical Evaluation Module (GEM), streamlines and automates selected SAPHIRE inputs and processes for performing event assessments.

SAPHIRE has also evolved with users' needs, and Versions 6 and 7 include new features and capabilities for developing and using larger, more complex models. For example, Version 7 can solve up to 2 million sequences and includes enhancements for cut set slicing, event tree rule linkage, and reporting options.

This NUREG-series report comprises seven volumes, which address SAPHIRE/GEM Versions 6 and 7. Volume 1, "Overview/Summary," gives an overview of the functions available in SAPHIRE and presents general instructions for using the software. Volume 2, "Technical Reference," discusses the theoretical background behind the SAPHIRE functions. Volume 3, "SAPHIRE Users' Manual," provides installation instructions and a step-by-step approach to using the program's features. Volume 4, "SAPHIRE Tutorial Manual," provides an example of the overall process of constructing a PRA database. Volume 5, "GEM/GEMDATA Reference Manual," discusses the use of GEM. Volume 6, "SAPHIRE Quality Assurance (QA) Manual," discusses QA methods and tests. Lastly, Volume 7, "SAPHIRE Data Loading Manual," assists the user in entering PRA data into SAPHIRE using the built-in MAR-D ASCII-text file data transfer process.

Christiana H. Lui, Director
Division of Risk Analysis
Office of Nuclear Regulatory Research

v

CONTENTS

EXECUTIVE SUMMARY

The U.S. Nuclear Regulatory Commission (NRC) has developed a powerful personal computer (PC) software application for performing probabilistic risk assessments (PRAs), called Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE).   Using SAPHIRE on a PC, an analyst can perform a PRA for any complex system, facility, or process. Regarding nuclear power plants, SAPHIRE can be used to model a plant's response to initiating events, quantify associated core damage frequencies and identify important contributors to core damage (Level 1 PRA). It can also be used to evaluate containment failure and release models for severe accident conditions, given that core damage has occurred (Level 2 PRA). It can be used for a PRA assuming that the reactor is at full power, at low power, or at shutdown conditions. Furthermore, it can be used to analyze both internal and external initiating events, and it has special features for transforming models built for internal event analysis to models for external event analysis.

SAPHIRE development and maintenance has been undertaken by the Idaho National Laboratory (INL). The INL began development of a PRA software application on a PC in the mid 1980s when the potential of PC applications started being recognized. The initial version, *Integrated Risk and Reliability Analysis System* (IRRAS), was released by the Idaho National Engineering Laboratory (now Idaho National Laboratory) in February 1987. IRRAS was an immediate success, because it clearly demonstrated the feasibility of performing reliability and risk assessments on a PC and because of its tremendous need (Russell 1987). The development of IRRAS continued over the following years, leading SAPHIRE to become a powerful PRA tool.

SAPHIRE provides the functions required for performing a PRA. Users can supply basic event data, create and solve fault trees and event trees, perform uncertainty analyses, and generate reports. Extending this analysis, GEM is a user-friendly interface that streamlines and automates select SAPHIRE inputs and processes for performing events assessments.

SAPHIRE contains editors or options for creating event trees and fault trees, defining accident sequences and basic event failure data, solving system fault trees and accident sequence event trees, quantifying cut sets, performing sensitivity and uncertainty analyses, documenting the results, and generating reports. SAPHIRE capabilities for performing a PRA are summarized below.

- Graphical fault tree construction
- Graphical event tree construction
- Rule-based fault tree linking
- Fast cut set generation
- Fault tree flag sets
- Failure data
- Uncertainty analysis
- Cut set editor, slice, display, and recovery analysis tools
- Cut set path tracing
- Cut set comparison
- Cut set end state partitioning
- End state analysis

- User-defined analysis types

SAPHIRE has a powerful report generation module. In addition, SAPHIRE has utility functions to recover a corrupted database, convert fault tree logic from alpha to graphics, and change user-defined constants. With SAPHIRE, a user can export the graphical fault trees and event trees to an industry-standard Windows metafile.

To use SAPHIRE, the user must create a project. A *project* is any logical grouping of fault trees and sequences with their associated basic events, cut sets, reliability data, and descriptions. Along with the project name, description, plant design, etc., it also contains cross-reference information for relationships between basic events, fault tree, sequence, event tree, end state, change sets, and gate in the project. The project table also contains the project sequence partition, and project sequence, fault tree, and end state recovery information along with the macro text. Once the project is created, the user can construct fault trees and event trees.

The Fault Tree Editor is used to create fault trees for the project. SAPHIRE creates an entry in the Fault Tree table for each fault tree created. SAPHIRE requires that certain conventions be used for naming fault trees. To use a fault tree as a top event in an event tree, or reference it in a sequence, the name must be specified in a special manner: (1) the name of the fault tree and the name of the top gate in the fault tree must be the same; (2) the name of the top event in the event tree and fault tree name must be the same.

SAPHIRE adds all the basic event references in the fault tree to the event table. Finally, SAPHIRE creates an entry in the event table for each transfer gate defined in the fault tree. This table maintains data for the fault tree attributes:

- Fault tree name and description
- Fault tree logic
- Minimal cut sets
- Top event probability
- Recovery rules
- Uncertainty analysis results.

The Event Tree Editor is used to generate the event trees for the project. When the user creates event trees using the Event Tree Editor, SAPHIRE automatically adds related information to the database. When the event tree is saved, an entry is created in the Event Tree table that contains the graphical information associated with the event tree. If the event tree contains an initiating event as the first top event in the tree, entries are automatically created in the Event Tree table. Entries include the name of the event tree, the initiating event for the event tree, and other descriptive data. Next, SAPHIRE creates an entry in the Fault Tree table for each top event defined in the event tree. By default, SAPHIRE assumes that each top event is represented by a fault tree with the same name. At the time the fault tree entry is created, SAPHIRE also creates a basic event in the Event table with the same name as the fault tree. The Event Tree table maintains data for the event tree attributes:

- Event tree name and description
- Event tree logic and graphic representation
- Initiating event
- Event tree macro, recovery and partition rules.

Once the fault trees are defined, one would enter the failure data for each basic event in the database. The Event table maintains data for the basic event attributes:

- Basic event names and description
- Event attributes
- Transformation information
- Uncertainty information
- Failure data
- Process flag.

All SAPHIRE calculations (e.g., cut-set generation and sensitivity analysis) use the data stored in the current case. The current-case data are created by applying change sets to base-case data (via the **Generate** option). Note that the base-case results *do not* change each time an analysis is performed. 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. Within the project, the logic models (fault trees and event trees) and associated graphics are stored separately from the base/current case data.

The event data in the current case can be manipulated, via change sets, to examine the changes in the probabilities of plant accidents and accident sequence failures based on basic events. A change set is a set of data modifications to be applied to basic events during fault tree or sequence analysis. Within a change set, the user can add events to the change set, modify selected event probabilities, or reset those event probabilities to the base case values. If the base-case data are changed and the Generate option is *not* performed, the data used for the analysis will not reflect the changes. However, if the user attempts to perform an analysis after making a data change, SAPHIRE will prompt the user to generate data changes.

The uncertainty analysis functions in SAPHIRE estimate the variability (due to the uncertainties in the basic event probabilities) of either a fault tree top event probability or an event tree sequence frequency. In a traditional uncertainty analysis, SAPHIRE samples the user-specified distributions for each basic event in a group of cut sets, then quantifies these cut sets using the sample values.

In addition to the fault tree and event tree analysis capabilities, SAPHIRE also allows users to gather minimal cut sets from the sequences into end states. End states are stand-alone groups of cut sets representing a specific outcome (e.g., core damage, large early release).

SAPHIRE has become a powerful PRA tool. Its relational database structure and editing rules offer the capability for sophisticated modeling of accident progression and, therefore, offer the means for a more accurate and extremely efficient analysis. Several other features, many constructed from feedback by users dealing with large-scale PRA models, make SAPHIRE among the fastest and most sophisticated PRA codes available today.

ACKNOWLEDGMENTS

ACRONYMS

ASP	Accident Sequence Precursor
DLL	Dynamic Link Library
ETG	event tree graphic
FEP	Fault Tree, Event Tree, and Piping and Instrumentation Diagram
GEM	Graphical Evaluation Module
INEL	Idaho National Engineering Laboratory
INL	Idaho National Laboratory
IRRAS	Integrated Reliability and Risk Analysis System
LERF	large early release frequency
LOCA	loss-of-coolant accident
LOOP	loss of off-site power
MAR-D	Models and Results Database
MGL	multiple Greek letter
NPP	nuclear power plant
NRC	Nuclear Regulatory Commission
P&ID	Piping and Instrumentation Diagram
PC	personal computer
PDS	Plant Damage State
PRA	probabilistic risk analysis
PWR	pressurized water reactors
QA	Quality Assurance
RAW	risk achievement worth
RCP	reactor coolant pump
SAPHIRE	Systems Analysis Programs for Hands-on Integrated Reliability Evaluations
SARA	System Analysis and Risk Assessment
SETS	Set Equation Transformation System
SPAR	Standardized Plant Analysis Risk
V&V	verification and validation

Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE)

Vol. 1 Summary Manual

1.   INTRODUCTION

1.1   Background

The U.S. Nuclear Regulatory Commission (NRC) has developed a powerful personal computer (PC) software application for performing probabilistic risk assessments (PRAs), called Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE).

Using SAPHIRE on a PC, an analyst can perform a PRA for any complex system, facility, or process. Regarding nuclear power plants, SAPHIRE can be used to model a plant's response to initiating events, quantify associated core damage frequencies and identify important contributors to core damage (Level 1 PRA). It can also be used to evaluate containment failure and release models for severe accident conditions, given that core damage has occurred (Level 2 PRA). It can be used for a PRA assuming that the reactor is at full power, at low power, or at shutdown conditions. Furthermore, it can be used to analyze both internal and external initiating events, and it has special features for transforming models built for internal event analysis to models for external event analysis. It can also be used in a limited manner to quantify risk for release consequences to both the public and the environment (Level 3 PRA). For all of these models, SAPHIRE can evaluate the uncertainty inherent in the probabilistic models.

SAPHIRE development and maintenance has been undertaken by the Idaho National Laboratory (INL). The INL began development of a PRA software application on a PC in the mid 1980s when the enormous potential of PC applications started being recognized. The initial version, *Integrated Risk and Reliability Analysis System* (IRRAS), was released by the Idaho National Engineering Laboratory (now Idaho National Laboratory) in February 1987. IRRAS was an immediate success, because it clearly demonstrated the feasibility of performing reliability and risk assessments on a PC and because of its tremendous need (Russell 1987). Development of IRRAS continued over the following years. However, limitations to the state of the-art during those initial stages led to the development of several independent modules to complement IRRAS capabilities (Russell 1990; 1991; 1992; 1994). These modules were known as Models and Results Database (MAR-D), System Analysis and Risk Assessment (SARA), and Fault Tree, Event Tree, and Piping and Instrumentation Diagram (FEP).

IRRAS was developed primarily for performing a Level 1 PRA. It contained functions for creating event trees and fault trees, defining accident sequences and basic event failure data, solving system fault trees and accident sequence event trees, quantifying cut sets, performing sensitivity and uncertainty analyses, documenting the results, and generating reports.

MAR-D provided the means for loading and unloading PRA data from the IRRAS relational database. MAR-D used a simple ASCII data format. This format allowed interchange of data between PRAs performed with different types of software; data of PRAs performed by different codes could be converted into the data format appropriate for IRRAS, and vice-versa.

SARA provided the capability to access PRA data and results (descriptive facility information, failure data, event trees, fault trees, plant system model diagrams, and dominant accident sequences) stored in MAR-D. With SARA, a user could review and compare results of existing PRAs. It also provided the

capability for performing limited sensitivity analyses. SARA was intended to provide easier access to PRA results to users that did not have the level of sophistication required to use IRRAS.

FEP provided common access to the suite of graphical editors. The fault tree and event tree editors were accessible through FEP as well as through IRRAS, whereas the piping and instrumentation diagram (P&ID) editor was only accessible through FEP. With these editors an analyst could construct from scratch as well as modify fault tree, event tree, and plant drawing graphical representations needed in a PRA.

Previous versions of SAPHIRE consisted of the suite of these modules. Taking advantage of the Windows 95 (or Windows NT) environment, all of these modules were integrated into SAPHIRE Version 6; more features were added; and the user interface was simplified.

With the release of SAPHIRE versions 5 and 6, INL included a separate module called the Graphical Evaluation Module (GEM). GEM provides a highly specialized user interface with SAPHIRE, automating SAPHIRE process steps for evaluating operational events at commercial nuclear power plants. In particular, GEM implements many of the accident sequence precursor (ASP) program analysis methods. Using GEM, an analyst can estimate the risk associated with operational events very efficiently and expeditiously.

1.2   Purpose

This report introduces and summarizes this NUREG/CR series, Volumes 1-7, documents that describe the functions of SAPHIRE and GEM—technical documents that describe why it functions the way it does and user's guides that describe how to use the individual functions. Detailed documentation and instructions for using the code available in the on-line reference manual have been reformatted to present in this set of NUREG/CR volumes.

In the past, detailed documentation was provided in hard copy format. A series of NUREG/CR reports was released documenting each individual module discussed above (NUREG/CR-6166; Volumes 1 through 8, 1994), document Version 5 of the SAPHIRE suite of codes. However, in response to continually increasing NRC needs for PRA capability and in response to feedback from users, SAPHIRE has undergone changes and enhancements. The consequent need for a continual update of detailed SAPHIRE documentation is met by the electronic on-line manual, which is updated with each major release of a SAPHIRE version. This new set of NUREG/CR reports described in this report reflects an updated snapshot of the documentation for the current release of SAPHIRE.

1.3   SAPHIRE Manual Overview

The following volumes that constitute this series describe SAPHIRE and its capabilities for performing a PRA. A brief summary of each volume follows below. More detailed descriptions can be found in the respective individual volumes.

- Overview/Summary (Volume 1)

    This volume summarizes this NUREG series and overviews the functions of SAPHIRE.

- Technical Reference (Volume 2)

    This volume presents the theoretical background behind the functions of SAPHIRE and includes the following topics: Set Theoretic Concepts, Review of Fault Tree Logic, Probability Concepts, Determination of Minimal Cut Sets, Quantification Tools for Probabilities and Frequencies, Event Probability Calculation Types, Importance Measures, Uncertainty, and Seismic Calculations.

Improvements include an expansion of Quantification Tools to include a section that describes the concept of "compound events" and how they work. This section will include discussions of the various plug-ins, including the common cause failure plug-ins. Each plug-in is described and the theoretical basis explained. In addition, sections describe the process of end state gathering and cut set partitioning and the cut set recovery process for both fault trees and sequences.

Improvements also include expansion of the Importance Measures to include a discussion on (a) uncertainty importance, (b) importance measures added to support users working in various risk-informed activities, and (c) the "importance wizard" capability, which allows a user to better estimate the risk achievement worth (RAW) of a component.

Improvements include expansion of the Uncertainty Analysis to include a more thorough discussion of the Latin Hypercube Sampling methodology available in SAPHIRE.

Expansion of the External Events section, which discusses fire and flood analysis in addition to seismic analysis, has also been updated.

- SAPHIRE Users' Manual (Volume 3)

  This manual is currently divided into 14 chapters and 5 appendices. The first two chapters introduce SAPHIRE, give installation instructions, and explain the basic features of SAPHIRE. The third chapter details SAPHIRE database concepts to assist in learning how SAPHIRE works. The remaining chapters present a step-by-step approach to using the features of SAPHIRE. Chapter 14 discusses the capabilities and limitations of SAPHIRE. The appendices cover some of the advanced features of SAPHIRE. Each of these chapters and appendices has been updated and expanded.

- SAPHIRE Tutorial Manual (Volume 4)

  This manual describes the overall process of constructing a PRA database using a simple modeling domain, that of driving a car. Many of the SAPHIRE's features are discussed as part of this PRA modeling. In addition, many of the "advanced" features in SAPHIRE are used, for example cut set recovery rules and template events.

- GEM/GEMDATA Reference Manual (Volume 5)

  This manual is currently divided into seven chapters and two appendices. The first two chapters introduce GEM, give installation instructions, and explain the basic features of GEM/SAPHIRE. The remaining chapters present a step-by-step approach to using the features of GEM. Each of these chapters and appendices will be updated and expanded as needed.

- SAPHIRE Quality Assurance (QA) Manual (Volume 6)

  This manual details the methods and tests used to provide QA. It also compares the SAPHIRE Testing, Validation, and Verification methodology against the IEEE Independent Validation and Verification methodology. Configuration control is discussed and testing examples are provided.

- SAPHIRE Data Loading Manual (Volume 7)

  This manual is currently divided into four chapters and four appendices. The first two chapters introduce SAPHIRE and important database concepts. The third chapter discusses a sample database, and the fourth chapter discusses the complete process of loading the sample database. Each of these chapters and appendices are updated and expanded as appropriate. Data conversion methods and issues, including validation, are discussed.

1.4   Overall SAPHIRE Capabilities

SAPHIRE capabilities for performing a PRA are summarized below. A more detailed description can be found in the individual sections of the detailed, on-line documentation.

> NOTE: This document assumes you are familiar with PRA methods and terminology. It is an overview of the SAPHIRE software. An on-line reference manual is available with installation of the software. The on-line manual does not instruct in the PRA methodology but is merely a reference guide for the SAPHIRE software.

- Graphical fault tree construction

  Fault trees are constructed and modified with an interactive and flexible graphical editor; or with a fault tree logic editor similar to the hierarchical file explorer available in Windows.

- Graphical event tree construction

  Event trees are constructed and modified with an interactive and flexible graphical editor, which allows linkage of multiple event trees and creation of very large event trees.

- Rule-based fault tree linking

  Accident sequences are generated using a linking module that uses a powerful set of user-defined rules to specify top-event substitutions. These rules allow testing the current state of the solution process at each branch point in the event tree.

- Fast cut set generation

  Extremely large fault trees and accident sequences are solved very quickly. SAPHIRE's solution engine is among the fastest in the industry.

- Conditional truncation

  Accident sequence cut sets can be generated using a "conditional truncation probability," whereby the initiator frequency is assumed to have a value of 1.0 during the initial truncation process.

- Fault tree flag sets

  "Flag sets" are included for pruning parts of fault trees to adjust their logic to different conditions. This feature saves you from having to create multiple copies of a fault tree that are slightly different from one another.

- Failure data

  Failure data are handled through a failure data module that includes a "template." A template allows multiple basic events to refer back to a single "reference" basic event.

- Uncertainty analysis

  Both Monte Carlo and Latin Hypercube methods are available for performing uncertainty analysis. An event uncertainty viewer displays the cumulative distribution or probability density curve.

- Importance measures

  The traditional importance measures such as Fussell-Vesely, Risk Increase Ratio or Interval, Risk Reduction Ratio or Interval, and Birnbaum are available. However, because these measures are not suitable for all types of analyses, the NRC developed extended importance measures for passive components analysis that include cumulative Fussell-Vesely, Fussell-Vesely/Birnbaum, and group importance. These measures have been incorporated into SAPHIRE.

In addition to the importance measure calculations, the uncertainty on these measures are now evaluated using Monte Carlo sampling techniques.

- Cut set editor and recovery analysis tools

  A rule-based editor is available for defining rules for modifying cut sets. These rules can be used to apply recovery on cut sets after an initial analysis has been performed. They can also be used to perform extensive changes to the cut set logic to incorporate new or different conditions.

- Versatile cut set slice and display

  A unique cut set "sort and slicer" is available to search and partition a large number of cut sets in fault trees, sequences, or end states.

- Cut set path tracing

  A "path tracer" is available to find the exact path through the logic that generated a particular cut set. As with other features contained within SAPHIRE, the speed of the "path tracer" is at the top of its class. This option will trace a cut set from an end state to its source in a fault tree.

- Cut set comparison

  A color-based cut set comparison utility is available for viewing changes between before-and-after lists of minimal cut sets.

- Cut set end state partitioning

  Cut sets can be partitioned into "end states" via a set of rules allowing each cut set to be assigned to a specific "end state."

- End state analysis

  An "end state analysis module" provides the ability for gathering and analyzing accident sequence end states.

- Fire/Flood/Seismic analysis

  SAPHIRE offers a powerful set of tools for simplifying the process of performing external event analysis if an internal event PRA is available. Fault trees built for internal event analysis can be transformed into external event-specific fault trees automatically. Furthermore, locations can be clustered into zones, and location-specific event trees can be transformed into zone-specific fault trees.

- User-defined analysis types

  You can define up to eight different analysis types for storing results. This is in addition to the eight fixed analysis types used by the software.

- Plug-in architecture

  A user-callable "plug-in" architecture allows interfacing to a customized library of probability calculation models. For example, shipped with SAPHIRE are plug-ins representing an alpha-factor common cause, a multiple Greek letter common cause, and a "supercomponent." In addition, advanced plug-in modules are available to determine corrosion rates from flow-accelerated corrosion and its corresponding impact on the pipe failure probability.

- Report generation module

  SAPHIRE has a powerful report generation module. You can create extensive reports containing any type of information you wish to include.

- Utility functions

  SAPHIRE has utility functions to recover a corrupted database, convert fault tree logic from alpha to graphics, and change user-defined constants.

- Graphical export to Windows metafiles

  With SAPHIRE, you can export the graphical fault trees and event trees to a Windows metafile. This allows importing the graphics into other Windows applications.

- International language support

  Enhancements to SAPHIRE allow for international text characters. For example, fault tree names can have a mixture of English and Russian. Consequently, logic gates or basic events with a name such as "ИКЗелйныЙ TREE" are permissible.

- Dual language support

  SAPHIRE has the capability to store two names, descriptions, and text fields for each entity in a project. You can switch the display between the primary and alternate information easily. This allows a project to contain English and another language for each entity.

- On-line context sensitive help

  SAPHIRE contains a help manual with over 500 pages of useful information.

Since the purpose of this document is to present a general overview of SAPHIRE, recent changes and enhancements may not be specifically addressed. Information on these changes and enhancements can be found within the electronic help file or on the SAPHIRE Web site at saphire.inl.gov.

Table 1 lists the available features of SAPHIRE version 6.x and SAPHIRE version 7.x.

Table 1 Features offered in released versions

Item	Description of Feature	Version 6.x	Version 7.x
A	Cut Set Sequence Generation		
A.1	Rule-based Fault Tree Linking	X	X
A.2	Linking of Small Tree Events	X	X
A.3	Linking of Large Tree Events	X	X
A.4	Sequence Capacity	64 thousand	2 million
B	Cut Set Generation		
B.1	Fault Trees	X	X
B.2	Event Trees	X	X
C	Cut Set Gathering		
C.1	Sequence End States	X	X
C.2	Sequence End State Cut Sets	X	X
D	Cut Set Partitioning		
D.1	End State Definition by rules	X	X
E	Cut Set Slice		
E.1	By Event	X	X
E.2	By Probability		X
E.3	By Rules		X
F	Cut Set Postprocessing (Recovery)		
F.1	Event Trees	X	X
F.2	Fault Trees	X	X
G	Change Sets		
	(Selected subset of Basic Events for temporary analysis)		
G.1	Single event selection	X	X
G.2	Multiple event selection	X	X
G.3	Group event selection	X	X
H	Flag Sets		
	(Selected subset of Basic Events with logic changes only)		
H.1	Cut Set Verification with Static Flag Sets	X	X
H.2	Cut Set Verification with Dynamic Flag Sets (linkage rules)	X	X
H.3	Applicable to Fault Trees	X	X
H.4	Applicable to Sequences	X	X
H.5	Applicable to Fault Trees within Sequences	X	X
I	Cut Set Quantification Methods		
I.1	Min-Cut	X	X
I.2	Min-Max	X	X
I.3	Rare Event	X	X
I.4	Split Fraction (sequences only)	X	X

Item	Description of Feature	Version 6.x	Version 7.x
J	Cut Set Analysis		
J.1	Cut Set Verification – cut sets solved, gathered, with truncation by size or probability, auto recovery	X	X
J.2	Cut Set path tracing	X	X
J.3	Cut Set comparison	X	X
J.4	Fault Tree	X	X
J.5	Event Trees / Sequences	X	X
J.6	End States	X	X
K	Basic Events (building of Relations)		
K.1	Basic Events – Generation	X	X
K.2	Basic Event – Templates	X	X
L	Basic-Event Calculations		
L.1	Compound Events Common-cause plug-in modules		
L.1.1	Common-cause alpha-factor module	X	X
L.1.2	Common-cause beta-factor module	X	X
L.1.3	Common-cause capacity load module	X	X
L.1.4	Common-cause multiple Greek letter module	X	X
L.1.5	Common-cause multiple group module		X
L.1.6	Common-cause alpha-staggered module	X	X
L.1.7	Loss-of-offsite power module		X
L.1.8	Time series module	X	X
L.1.9	General calculation module		X
L.2	Failure Probability on Demand	X	X
L.3	Failure Probability to Run	X	X
L.4	Value input (for any value)		X
L.5	Failure Probability to Run w/ repair	X	X
L.6	Failure Probability to Run	X	X
L.7	House Event True (Prob = 1.0), i.e. failed	X	X
L.8	House Event False (Prob = 0.0), i.e. success	X	X
L.9	House Event Ignore	X	X
L.10	Compound Event		X
L.11	Human Factor Event		X
L.12	Fault tree Min Cut Upper Bound Value	X	X
L.13	End State Min Cut Upper Bound Value	X	X
L.14	Ground Acceleration Value	X	X
L.15	Hazard Curve	X	X
M	Importance Measures		
M.1	Fussell-Vesely Importance Measure	X	X
M.2	Birnbaum Importance Measure	X	X
M.3	Risk increase ratio importance measure	X	X
M.4	Risk reduction ratio importance measure	X	X
M.5	Risk increase interval importance measure	X	X

Item	Description of Feature	Version 6.x	Version 7.x
M.6	Risk reduction interval importance measure	X	X
M.7	Group importance measure	X	X
M.8	Uncertainty determination on Importance Measures		X
N	Model Creation		
N.1	Fire and flooding transformation capability	X	X
N.2	Fault Tree text editor	X	X
N.3	Fault Tree graphical editor	X	X
N.4	Event Tree text editor	X	X
N.5	Event Tree graphical editor	X	X
N.6	Improved Logical Editors		X
N.7	Basic Load / Extract Data Models	X	X
N.7.1	Extract All		X
N.7.2	Load All		X
N.7.3	Extract All File types		X
N.7.4	Load All / Group		X
N.7.5	Fault Tree Logic	X	X
N.7.6	Designate output folder location		X
N.8	Graphical Export to Windows metafiles	X	X
N.9	Database Recovery	X	X
N.10	Database MAR-D Load and Extract	X	X
N.10.1	Event Tree MAR-D	X	X
N.10.2	Fault Tree MAR-D	X	X
N.10.3	Basic Event MAR-D	X	X
N.11	Alternate names and descriptions for basic events only	X	X
N.12	Alternate names and descriptions for all database objects (for multilingual use)		X
N.13	Model Version Upgrade (backward compatible only)	X	X
O	Model Creation Logic Gate Types		
	(Maximum inputs 256 unless otherwise specified)		
O.1	AND	X	X
O.2	OR	X	X
O.3	N of M (Max N=98 , Max M=99)	X	X
O.4	NAND (Not AND)	X	X
O.5	NOR (Not OR)	X	X
O.6	Transfer Gate	X	X
O.7	Inhibit gate	X	X
P	Uncertainty Calculations		
	(Monte Carlo and Latin Hyper Cube Sampling)		
P.1	None (or Point Value only)	X	X
P.2	Normal Distribution	X	X
P.3	Lognormal Distribution	X	X

Item	Description of Feature	Version 6.x	Version 7.x
P.4	Beta Distribution	X	X
P.5	Chi Squared Distribution	X	X
P.6	Exponential Distribution	X	X
P.7	Uniform Distribution	X	X
P.8	Constrained non-informative Distribution	X	X
P.9	Gamma Distribution	X	X
P.10	Maximum Entropy Distribution	X	X
P.11	Dirichlet Distribution	X	X
P.12	Seismic Log Normal analysis	X	X
P.13	Histogram Distribution	X	X
P.14	Triangular Distribution		X
Q	Uncertainty Calculations (Parameter Settings)		
Q.1	Seed	X	X
Q.2	Sample Size	X	X
R	General Support Features		
R.1	Sensitivity Wizard		X
R.2	Importance Measures Wizard		X
R.3	Embedded Macro capability		X
R.4	Editing User Information		X
R.5	Page numbering control on graphic format	X	X
R.6	Conversion from alpha to graphic format	X	X
R.7	On-line Context Sensitive help		X
R.8	Parallel processing (Linux only)		X
S	General Support Features (Report Generation)		
S.1	Project Reports	X	X
S.2	Attributes	X	X
S.3	Basic Event	X	X
S.4	Fault Tree	X	X
S.5	Event Tree	X	X
S.6	End State	X	X
S.7	Sequence	X	X
S.8	Change Set		X
S.9	Flag Set		X
S.10	Gate	X	X
S.11	Histogram	X	X
S.12	Slice		X
S.13	User Info		X
S.14	Cross Reference Reports	X	X
S.15	Cross Reference Reports from single location		X
T	Report Format Types		
T.1	ASCII	X	X
T.2	RTF		X
T.3	HTML		X

Item	Description of Feature	Version 6.x	Version 7.x
U	General Analysis Types		
U.2	Initiating Event Analysis	X	X
U.3	Condition Assessment Analysis	X	X
U.4	Accident Sequence Precursor	X	X
U.4.2	User defined analysis types	X	X
U.4.3	Load-capacity calculation module		X
V	Application Program Interface		
V.1	Microsoft Visual Basic and VBA interface	X	X
V.2	Microsoft Visual C\C++ interface	X	X
V.3	Borland Delphi		X

1.5   GEM Overall Capabilities

GEM is a special application designed to evaluate operational occurrences using the NRC's Standardized Plant Analysis Risk (SPAR) models and methods. GEM simplifies the process of modifying basic events in the models (e.g. failed or out-of-service components) and automates the calculation of CCDP. The three assessment types that GEM automates are as follows:

1. Initiating Event Assessments

   GEM automatically applies new recovery action failure probabilities to applicable basic events, depending on the initiating event being analyzed, and calculates a new CCDP based on the new probabilities and any user-defined changes.

2. Conditional Event Assessments

   This assessment type is used for conditions not involving an initiating event. GEM calculates a CCDP based on user-modified basic event probabilities and the duration of the condition.

3. Level 2/3 Analyses

   GEM automates the process of using PRA models that contain Level 2 Large early release frequency (LERF) models.

To use its full capabilities, GEM uses SAPHIRE plant models designed specifically for ASP evaluations. Therefore, GEM requires specific user-defined data and will not use special GEM-developed analysis capabilities with models not formatted to the SPAR model specifications.

2. GETTING TO KNOW SAPHIRE

This section explains how to install the SAPHIRE software and how to start and quit the application, gives pointers on how to get around inside the software easily, and discusses some of the special features of the software.

2.1 Installation

2.1.1 Software and Hardware Requirements

- Windows NT 4.0, 2000, XP

- Pentium III or higher, IBM-PC compatible

- 64 MB RAM for Windows NT

- 15 MB free disk space (minimum for installation)

NOTE: The total disk space requirement depends on the number and size of the projects created.

2.1.2 Installation Instructions

SAPHIRE can be installed from a CD-ROM or via a download from the SAPHIRE Internet site http://saphire.inl.gov.

To install from CD-ROM:

1. Insert the CD-ROM.

2. From the Windows Start Bar, click Run.

3. Type D:\SAPHIREI.EXE (where D: is the CD-ROM drive name).

4. The SAPHIRE Setup program prompts you through the installation process. Follow the instructions on the screen.

2.2 Using SAPHIRE

2.2.1 Starting SAPHIRE

1. From the Windows Start Bar, click **Programs**.

2. Select SAPHIRE for Windows.
    OR
   From the Windows desktop, double click the SAPHIRE program icon.

3. Proceed by selecting a project with which to work. See the on-line help manual for details on how to select a project.

2.2.2 Exiting SAPHIRE
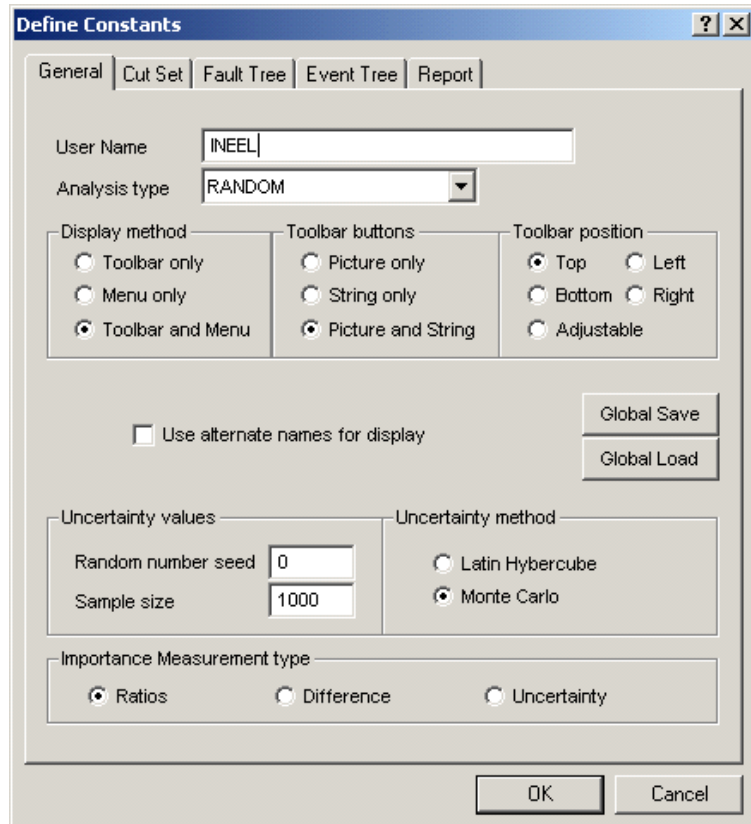
1. Select **File** from the menu bar.

2. Select the **Exit** submenu option. The SAPHIRE database will close and the program will terminate.

2.2.3 Using SAPHIRE for the First Time

When SAPHIRE for Windows is invoked for the first time after installation, the constants dialog will be displayed. This dialog allows you to enter default values desired for various functions within SAPHIRE. Default values are provided in most of the fields. This dialog will not appear on subsequent entries into SAPHIRE.

1. The first time SAPHIRE for Windows is run after installation, the *Define Constants* dialog will appear.

2. Fill in and change the information as appropriate.

3. When complete, choose the **OK** button.

4. The SAPHIRE window will appear, and you can proceed.

To change defaults, this dialog can be called up from the **Utility | Define Constants** option. To view the definition or description of the various options and buttons, click on the question mark in the upper-right corner of the dialog, and then select the option in question.

2.2.4 Converting from Previous SAPHIRE (IRRAS) Versions

This section describes the procedure to follow when a previous version of SAPHIRE (IRRAS, SARA, GEM, etc.) has been used. When SAPHIRE for Windows is invoked with an older database, the *Version Conflict* dialog will appear. Choose the **Yes** button to continue. The data for the current project will be converted to the most recent SAPHIRE version format.

NOTE: Once the data have been converted to an updated version, they cannot be used with older versions of SAPHIRE (e.g., databases converted or created in 7.x cannot be used in version 6.x).

After conversion is complete, the SAPHIRE window will appear and you can proceed.

2.3   Getting Around Within SAPHIRE

2.3.1   About List Boxes in SAPHIRE

Many dialogs in SAPHIRE contain list boxes. In some list boxes, multiple items can be selected for processing. The following discussion offers some hints on selecting list box items.

An item in a list box is "selected" if it is highlighted, that is, displayed in a color other than the other items in the list. The illustrations in this manual show selected list box items highlighted with a black background and white foreground characters, while the deselected items are displayed with a white background and black foreground characters. In the following example, the fault tree ECS is "selected."
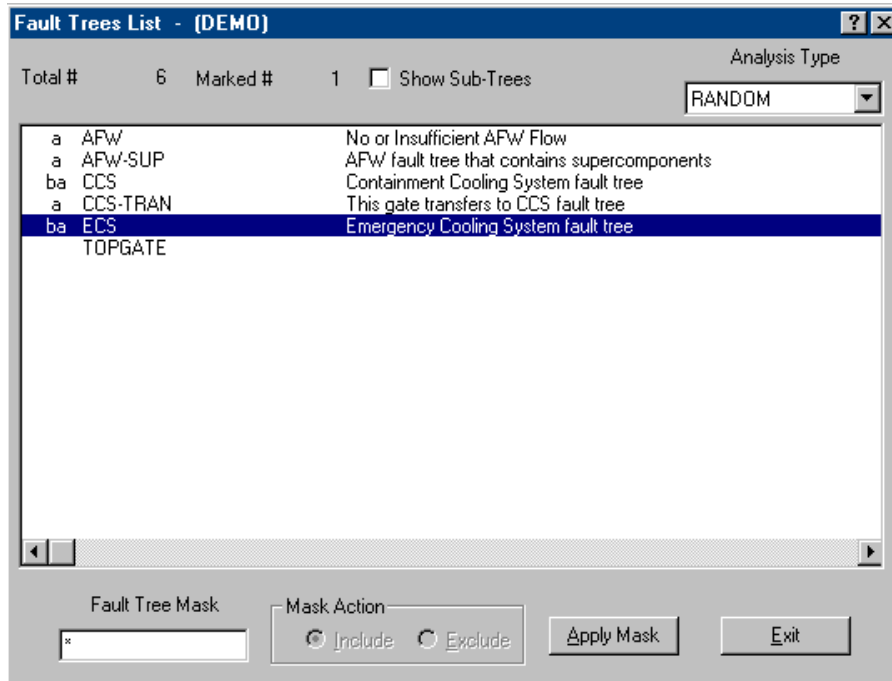
There are various ways to select items in a list. To select a single item, click on the desired item so that it is highlighted. To select multiple contiguous items, click and hold the mouse button down on the first desired item in the list and drag up or down the list to the last desired item. Then release the mouse button. Or, click the first desired item; then, holding down the **Shift** key, click the last desired item. To select multiple noncontiguous items in the list, click several desired items while holding down the **Ctrl** (control) key. To select all items in the list, highlight the first item then press the **Shift** and **End** keys, or press **Ctrl-A**.

2.3.2   Selecting List Items Using the Mask Feature

Some dialogs in SAPHIRE with list boxes provide a "Mask" capability, which allows you to select items from the list based on matched criteria. The following example shows the *Fault Tree Mask* entry field with the *Include* radio button selected.
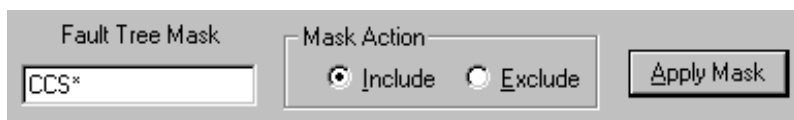
Generally, the mask is applied to the name of the item (e.g., Fault Tree name, or Event Tree name). The wildcard characters asterisk (*) and question mark (?) can be used in the mask. The asterisk represents

15

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.

To use the mask capability, a dialog with a *mask entry field* must be active.

1. With an appropriate dialog active, in the *mask entry field* enter the common characters of the names to be matched, including any wildcard characters, if desired.

2. In the *Mask Action* section, choose either the **Include** or **Exclude** radio button, depending on whether these items are to be included in the selection or excluded from it.

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

Any of the available functions on the active dialog can now be performed.

For specific information on using the sequence mask feature, see the topic in the SAPHIRE on-line reference manual.

2.3.3   Invoking Pop-up Menus

Many dialogs in SAPHIRE have pop-up menus available. Once the dialog has been displayed, invoke the pop-up menu by right-clicking. While holding down the right-hand side mouse button, select the desired menu option by highlighting it and then releasing the mouse button. Pop-up menus from any one dialog may vary, especially if the dialog contains a multiple-select list box. The options in a pop-up menu may depend on whether or how many items are selected.
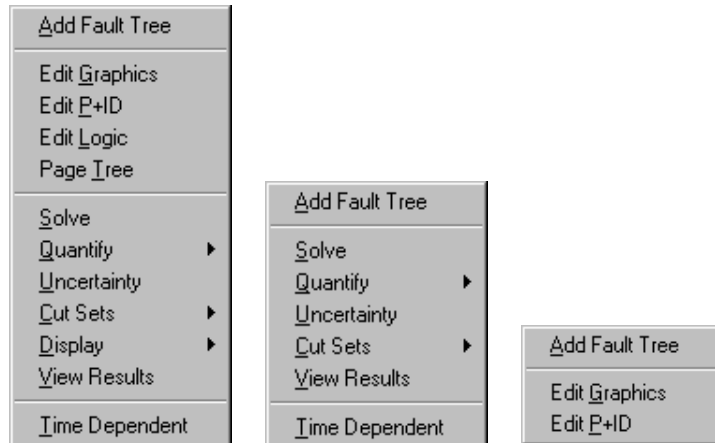
The three examples show the *Fault Trees List* dialog pop-up menus with one list item selected, more than one list item selected, and no list items selected.

2.3.4   SAPHIRE Toolbars

The SAPHIRE toolbar provides single-click access to the common SAPHIRE functions.

Depending on what function is currently selected, the toolbar will change to reflect available options. When the current dialog contains a multiple-select list box, some of the toolbar options may be disabled, depending on the number of list items selected. During processing, such as cut set generation, only the **Help** button is available.

The example below shows the main SAPHIRE toolbar:

The toolbar can be displayed or hidden by selecting the desired display method on the Define Constants dialog. Toolbar buttons can also be configured to display the bitmap (picture), text, or both.

16

2.3.5   Selecting the Analysis Type

Many dialogs in SAPHIRE have an Analysis Type drop-down list. This drop-down list allows you to select from eight pre-defined and eight user-defined analysis types.

Select an analysis type by clicking on the down arrow on the right side of the list and scrolling through the list until the desired analysis type is displayed. Then click on the desired analysis type. In the example here, *Random* is the current analysis type. The list has been scrolled about halfway down, and the *User1* analysis type is currently highlighted.
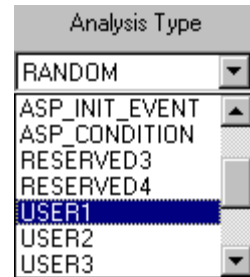
By selecting one of the analysis types from the drop-down list, the type of analysis to be performed on the selected fault tree(s), sequence(s), or end state(s) is indicated.

2.4   SAPHIRE Help

Included with the SAPHIRE installation is an on-line hypertext help system containing a complete user's guide to SAPHIRE features, along with a technical reference manual.

In general, help topics in the SAPHIRE help system are limited to one full screen of information, with links to related or more detailed information. Ordinarily, the topic pages contain a "PURPOSE" section that describes the selected topic and its primary function. After the "PURPOSE" section, a "STEPS" section provides systematic instructions for performing the function. Often, a picture of a dialog is included on the topic page. Brief explanations of the options on the dialog are provided below the graphic image, and when necessary, links for additional details are presented. On occasion, the graphic image on the topic page will contain hot-spots that provide links to other topic pages or invoke a pop-up window containing a definition. When applicable, a "SEE ALSO" section is included for links to related topics.

The example demonstrates the general layout of the topic pages in the SAPHIRE help system:

Links in topic text and on graphic images (also known as hot-spots) are indicated when the cursor changes to 🖑 . In addition, hot-spot text is displayed in green.

The help system can be accessed via the **Help** option from the menu or through context-sensitive help on each dialog. The **Help** menu offers access to the on-line hypertext help system through the **Contents Tab** or **Index Tab**. The **Contents Tab** provides access to help topics ordered by books and chapters. The **Index Tab** provides access to help topics through an alphabetical index based on key words.

Context-sensitive help provides information about the selected control on the active dialog box. It is accessed by choosing the **Help** button, ⍰ , located in the upper right-hand corner of the dialog.

In addition to the context-sensitive help, Tool Tips help is available on most dialogs. To access Tool Tips, position the cursor over a control on the active dialog. A small box will appear containing a brief description about that control.

2.5 User support

2.5.1 NRC Training

The INL offers training classes in the use of SAPHIRE for NRC users. Contact the NRC Program Manager for training information. Additional information about SAPHIRE training can be found on the SAPHIRE Web site at https://saphire.inl.gov, or by contacting the INL SAPHIRE program manager (Curtis Smith, Curtis.Smith@inl.gov) or the lead developer (Ted Wood, Ted.Wood@inl.gov).

2.5.2 SAPHIRE Users' group

The SAPHIRE Users' group is designed to provide quality technical support for the SAPHIRE software along with regular updates and technical information.

Membership benefits include:

- Software updates during the membership period

- Current software manuals and documentation

- Software consulting services

- Subscription to the *SAPHIRE FACETS* newsletter

- Full access to the SAPHIRE web site user group area

- User group training.

To join the Users' group, go to the SAPHIRE web site (http://saphire.inl.gov).

3. DATABASE CONCEPTS

Before SAPHIRE can be used effectively, an understanding of the SAPHIRE database structure and concepts is required. A relational database is used to manage inputs and outputs. SAPHIRE contains nine permanent tables, or relations, used to store and retrieve specific PRA information. For each table, three files exist:

*.DAT - This file contains the actual data and information about which data constitutes a "key." A "key" is used for searching and locating information.

*.IDX - This file contains indices into the data for keyed fields.

*.BLK - This file contains variable length data (such as cut sets) that are accessible through pointers contained in the *.DAT file.

A SAPHIRE database is called a *project*. Information for the project is divided into two separate parts, referred to as base case and current case. As delivered, SAPHIRE provides a standard project called DEMO. Access to any portion of the database is obtained through the appropriate project. Normally, a user will need to create a project for each PRA they are performing. All the information in a project is stored together in a folder (or subdirectory).

NOTE: Only one project may be stored in an individual folder on the PC's hard drive.

The following describes the steps a user would ordinarily take to perform a new PRA analysis using SAPHIRE. Certain information records are automatically added to the database depending on the functions performed by the user. Links are provided for detailed information about how the relations are used in each step:

1. Create a new project where the results of the PRA can be stored. Once the "project" is created, the user is ready to add information to the database.

2. Adjust the project operating constants (via the **Utility | Define Constants** option) if desired.

3. Create the event trees for the Project.

4. Create the associated fault trees.

5. Enter the failure data for each basic event in the project.

6. Generate the sequence entries for each valid sequence in the event trees.

7. Perform analysis, varying failure data using change sets.

3.1 Project Table

A *project* is any logical grouping of fault trees and sequences with their associated basic events, cut sets, reliability data, and descriptions. Each project is contained within a separate folder.

Information about the project is stored in the project table. The project table contains general information about the PRA. Along with the project name, description, plant design, etc., it also contains cross-reference information for relationships between basic events, fault tree, sequence, event tree, end state, change sets, and gate in the project. The cross-reference maps are updated automatically each time one of these pieces of data is modified. The project table also contains the project sequence partition, and project sequence, fault tree, and end state recovery information along with the macro text.

3.2   Fault Tree Table

The Fault Tree Editor is used to create fault trees for the project. SAPHIRE creates an entry in the Fault Tree table for each fault tree created. SAPHIRE requires that certain conventions be used for naming fault trees. To use a fault tree as a top event in an event tree, or reference it in a sequence, the name must be specified in a special manner: (1) the name of the fault tree and the name of the top gate in the fault tree must be the same; (2) the name of the top event in the event tree and fault tree name must be the same.

SAPHIRE adds all the basic event references in the fault tree to the Event table. If a description is provided for a basic event, then this description is also added to the database. All other information for the basic event is left unchanged or set to its default values.

SAPHIRE also creates an entry in the Gate table for each gate defined in the fault tree. The name, type, and description of each gate contained in the fault tree are stored in the database. This information can be edited. However, the changes are only reflected in the graphical fault trees if the user performs an "alpha to graphics" conversion on the fault trees that use the gate.

Finally, SAPHIRE creates an entry in the Event table for each transfer gate defined in the fault tree. This is done to allow the user to specify whether a transfer gate is to be expanded or whether the basic event representing the transfer gate is to be used.

This table maintains data for the fault tree attributes:

- Fault tree name and description
- Fault tree logic
- Graphic representation
- Minimal cut sets
- Top event probability
- Default flag set
- Default quantification method
- Recovery rules
- Importance measures
- Uncertainty analysis results.

3.3   Sequence Table

The Sequence table contains information about the accident sequences for an event tree. The user normally generates the sequence entries using the **Link Trees** option. When the user selects this option, SAPHIRE generates a sequence entry for each valid sequence in the event tree. Any sequences that may have been generated previously for this same event tree are automatically overwritten with the new valid sequence.

Each sequence contains an end state name that is created in the End State table if it does not already exist. Changing the name of an end state in a sequence only changes the name for that specific sequence and does not change the information in the End State table or the event tree for that sequence. When a sequence is solved, the cut sets generated are stored in the Sequence table.

This table maintains data for the accident sequence attributes:

- Name and description

- Logic (combination of successful and failed fault trees)

- Minimal cut sets

- Sequence probability

- Default flag set

- Default quantification method

- Sequence end state

- Associated event tree

- Partition and recovery rules

- Importance measures

- Uncertainty analysis results.

3.4   Event Tree Table

The Event Tree Editor is used to generate the event trees for the project. When the user creates event trees using the Event Tree Editor, SAPHIRE automatically adds certain information to the database.

When the event tree is saved, an entry is created in the Event Tree table that contains the graphical information associated with the event tree. If the event tree contains an initiating event as the first top event in the tree, entries are automatically created in the Event Tree table. Entries include the name of the event tree, the initiating event for the event tree, and other descriptive data.

Next, SAPHIRE creates an entry in the Fault Tree table for each top event defined in the event tree. By default, SAPHIRE assumes that each top event is represented by a fault tree with the same name. At the time the fault tree entry is created, SAPHIRE also creates a basic event in the Event table with the same name as the fault tree. This event is used when the fault tree logic is not expanded during fault tree analysis or when the logic has not yet been developed.

The Event Tree table maintains data for the event tree attributes:

- Event tree name and description

- Event tree logic and graphic representation

- Initiating event

- Event tree macro, recovery and partition rules.

3.5   Event Table

Once the fault trees are defined, you need to enter the failure data for each basic event in the database. You can do this with either the **Modify** option or with the **Generate** option and **Base Case Update** options.

If you use the **Modify** option to change the default failure data, then be sure to use the **Generate** option to generate the current failure data to be used in the remainder of the analysis. Any time changes are made to the basic event failure data, you MUST use the **Generate** option before the changes can be used. If it appears that changes made to basic event failure data are not being used properly by SAPHIRE, you should check to ensure the proper data were generated and saved.

The Event table maintains data for the basic event attributes:

- Basic event names and description
- Event attributes
- Transformation information
- Uncertainty information
- Failure data
- Process flag.

3.6   Change Set Table

A change set is a set of data modifications to be applied to basic events during fault tree or sequence analysis.

The Change Set table stores the name, description, and event changes for change sets and flag sets created for this project. This information can be edited from the **Modify** option, or for change sets, from the **Generate** option.

This table is maintained to track event changes relating to event class and event probability.

3.7   User Information Table

The User Information table contains basic operating constants for SAPHIRE. You may change these constants to reflect desired defaults. These data are unrelated to trees, but can be specified global or for the current project. Information stored includes cut set generation cutoff values, event tree and fault tree graphical editor preferences, and menu and toolbar defaults.

3.8   Analysis Type Table

SAPHIRE allows you to store multiple analysis types in the same project. For instance, you can perform a normal random analysis and then, using the same database, perform a fire or flood analysis. Each analysis uses the same fault trees and failure data, but stores the cut sets and other results in a different location in the database. This ability allows the user to track all the results from a comprehensive analysis in the same database.

In SAPHIRE, there are eight reserved analysis types and eight user-definable types. The first six reserved types are "Random," "Fire," "Flood," "Seismic," "Event Assessment," and "Condition Assessment." The remaining two reserved types are undefined. The user-definable types can be used to store results for any analysis the user desires. For example, one of the user-definable types can be used to store the results of a Low Power Shutdown analysis.

3.9   Base versus Current Case

Base case and *current case* refer to two separate parts of a *project* database. The base case data are stored in the database files as a "permanent" record. The current case data are used to perform an analysis (e.g., cut set generation and quantification).

All SAPHIRE calculations (e.g., cut-set generation and sensitivity analysis) use the data stored in the current case. The current-case data are created by applying change sets to base-case data (via the **Generate** option).

NOTE: The current-case results are overwritten each time a specific analysis is performed. The base-case results *do not* change each time an analysis is performed.

Current-case data can equal the base case data in order to reproduce the original study stored in the base case; however, only current-case cut sets can be modified with the cut-set editor. Changes made to cut sets with the cut-set editor are implemented automatically. A cut-set update is necessary if changes are made that create non-minimal cut sets.

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.

Within the project, the logic models (fault trees and event trees) and associated graphics are stored separately from the base/current case data. In addition, the change and flag sets are stored apart from the analysis data.
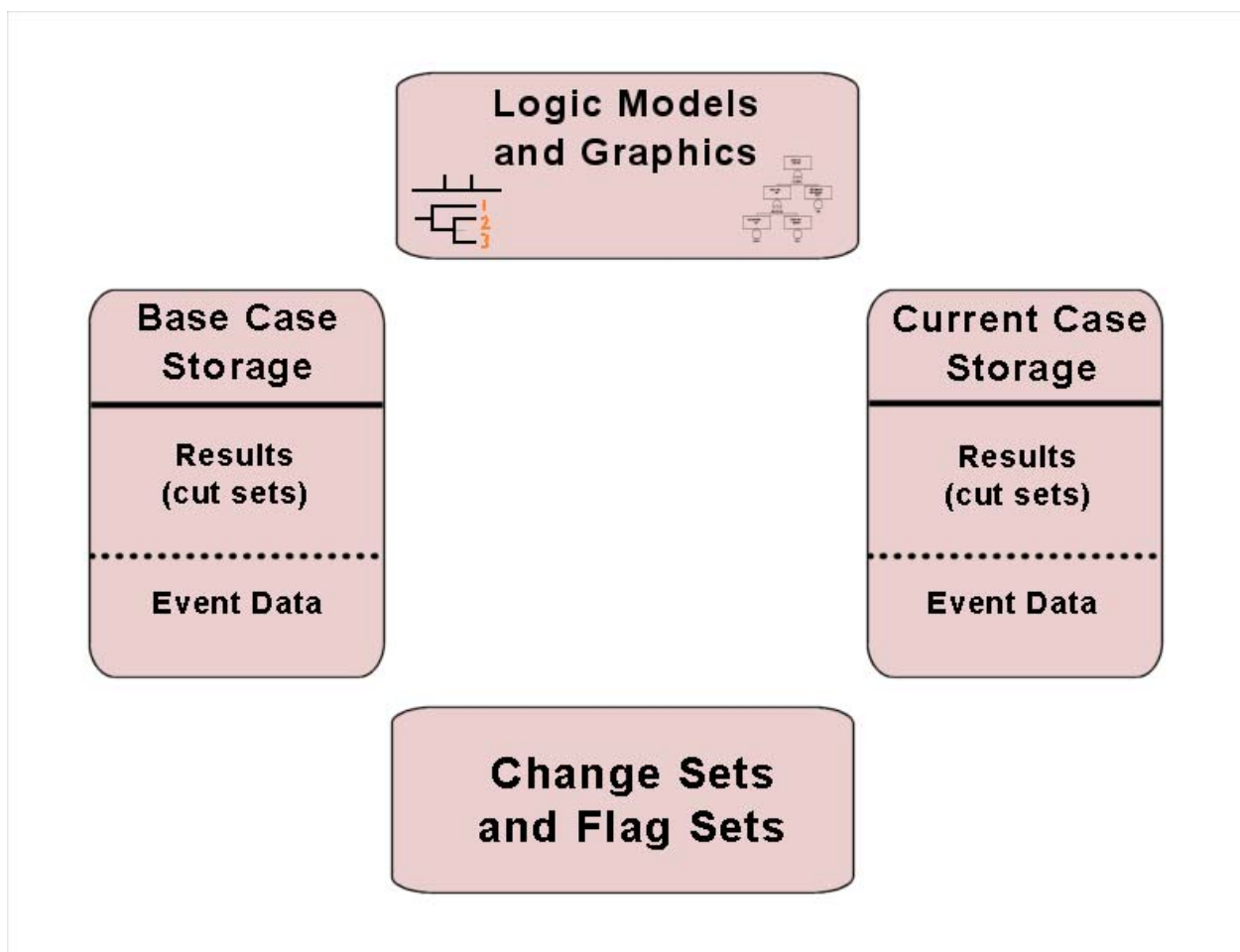
4.   GENERATING EVENT DATA

The Generate option is used to create a working copy of user event data for use in a particular analysis. These data can then be modified using change sets or used to update the base case data. SAPHIRE always uses current-case data for analysis.

The event data can be manipulated, via change sets, to examine the changes in the probabilities of plant accidents and accident sequence failures based on basic events. A change set is a set of data modifications to be applied to basic events during fault tree or sequence analysis. Change sets modify the current-case basic event data. A change set consists of both single and group changes for a list of basic events.

Within a change set, the user can add events to the change set, modify selected event probabilities, or reset those event probabilities to the base case values. The probability of events occurring can be recalculated. The user can construct change sets to be applied to the basic events for later propagation through existing cut sets. In addition, the user can perform a "base case update" that takes all current basic-event data and uses them to overwrite the base-case data. Note that the original base-case data are overwritten in this process.

If the base-case data are changed and the Generate option is *not* performed, the data used for the analysis will not reflect the changes. However, if the user attempts to perform an analysis after making a data change, SAPHIRE will prompt the user to generate data changes.

4.1   Change-Set Marking

Change sets are "marked" to determine which change sets will be used to generate current-case data. Order of marking determines which event changes take precedence.

To mark a change set, double-click the change set. If the change set was not previously marked, a numeric value (1–99) will appear to the left of the change set name, indicating the order of marking. The change sets in the list are rearranged by marked order, then alphabetically by type: first by condition assessment, then initiating event assessment type, finally by ordinary change set type.

If the change set was previously marked, it is unmarked (i.e., the numeric value is removed) and it is repositioned in the list.

If no change sets are marked when the Generate option is selected, then the current case data will be initialized to the base-case data.

4.2   Change-Set Processing

During change-set processing (i.e., generating event data or updating base-case values), three criteria are evaluated: probability changes, class changes, and the order of marked change sets.

If an event in a marked change set has both a class change and a probability change associated with it, the probability change takes precedence over the class change if a discrepancy arises.

If more than one change set is marked, the probability *and* class changes in the change sets marked with the highest number take precedence over any changes from lower numbered change sets (Set 2 is higher than Set 1; 3 is higher than 2). For example, suppose that Event A has a probability change associated with it in Change Set 1 and a class change associated with it in Change Set 2. The *class* change from Change Set 2 overwrites the *probability* change from Change Set 1.

The following example illustrates the outcome of processing three marked change sets with four events:

- Event A contains both a probability change and a class change associated with it in Change Set 2.

- Event B contains a probability change in Change Set 1, and a class change in Change Set 3.

- Event C contains only a class change in Change Set 1.

- Event D contains no probability changes or class changes associated with it in any of the three marked change sets.

The outcome of the Generate operation on these three marked change sets would be as follows:

- The current case data for event A would be set to the values associated with the probability change in Change Set 2. (Probability changes take precedence over class changes when they occur for the same event within the same change set.)

- The current case data for event B would be set equal to the values associated with the class change in Change Set 3. (The probability or class changes in higher numbered change sets take precedence over changes in lower numbered change sets.)

- The current case data for event C would be set to the values associated with the class change in Change Set 1.

- The current case data for event D would be set to the base case values for event D. (There were no probability changes or class changes made to event D in any of the marked change sets.)

4.3   Single Changes

This function provides the flexibility for experimentation with individual basic-event failure, uncertainty, seismic fragility data, and process flags. These data values may be set for a single event or for a selected group of events. This current data can then be applied to the specified event(s) for use during the analysis.

The probability changes made are reflected on the *Selected Events* dialog by showing a "P" to the left of the event name. Probability changes have a higher priority than class changes. When both are applied to an event, the probability change will be used during analysis.

The single change-set type also provides a reset option to set data values back to the base-case values.

4.4   Class Changes

This option allows the user to change event data parameters for a specified grouping of events in a change set. The event class is defined by entering search criteria in the mask fields. The more of these fields that are filled in, the finer the class definition becomes.

It is important to remember that probability changes have a higher priority than class changes. When both are applied to an event, the probability change will be used during analysis.

4.5 Generating Changes

The Generate function applies the event data modifications specified by the marked change sets to the current-case basic event data. This option must be executed before any data analysis if the change-set modifications are to be reflected in the analysis results. The Generate option creates new current-event data, which are used to perform an analysis.

If an event in the change set has both a class change and a probability change associated with it, the probability change takes precedence over the class change when generating new current-case event data values. If the event does not have an associated probability change or class change, then the current-case event data values will be set equal to the base-case data for that event.

4.6 Reporting Change-Set Data

Reports can be created that reflect the event modifications that currently exist within the basic-event change data file. There are three report types: Unaffected Events, Affected Events, and All Events. Each of the report types may be sorted by event name, ascending probability, or change sets. In addition, a report may be routed to the Report Viewer, a printer, or a disk file.

The report generated by this option shows the basic-event name and current values for calculation type, lambda value, tau value, mission time, probability value, change set name, and type of change applied.

5. FAULT TREES

Fault tree analysis allows the user to build, edit, and analyze fault tree models. A fault tree is a model of a system suitable for probabilistic analysis. The model is usually tailored for study of the system's failure.

When the fault tree toolbar button or the fault tree menu option is selected, the *Fault Trees List* is displayed. This dialog shows all top-level fault trees in the current project. A "c" in front of the fault tree name indicates that only current-case cut sets exist; a "C" indicates that both current-case cut sets and base-case cut sets exist. In addition, in SAPHIRE 7.x, a "G" in front of the fault tree name indicates that logic and graphics exist for the fault tree. A "g" indicates that only graphics exist for the fault tree, and an "L" indicates that only logic exists for the fault tree. From this dialog, the user can add a new fault tree record to the database or select existing fault tree(s) using the mouse or the Mask feature. Fault tree values can be recalculated after events and/or cut sets have been modified. The fault tree logic can be edited in alphanumeric format and organized into logical pages. In addition, uncertainty analyses can be performed, and importance information about basic events can be analyzed. Results can be displayed and reported in various formats.

5.1 Fault Tree Graphics

The Fault Tree Editor constructs fault tree diagrams, that is, a graphical representation or picture of the fault tree model built from fault tree objects. The fault tree logic structure can consist of any combination of the logic symbols that do not result in a logical loop. Fault tree logic can be thought of as a group or set of equations that describe the possible failure combinations of the system.
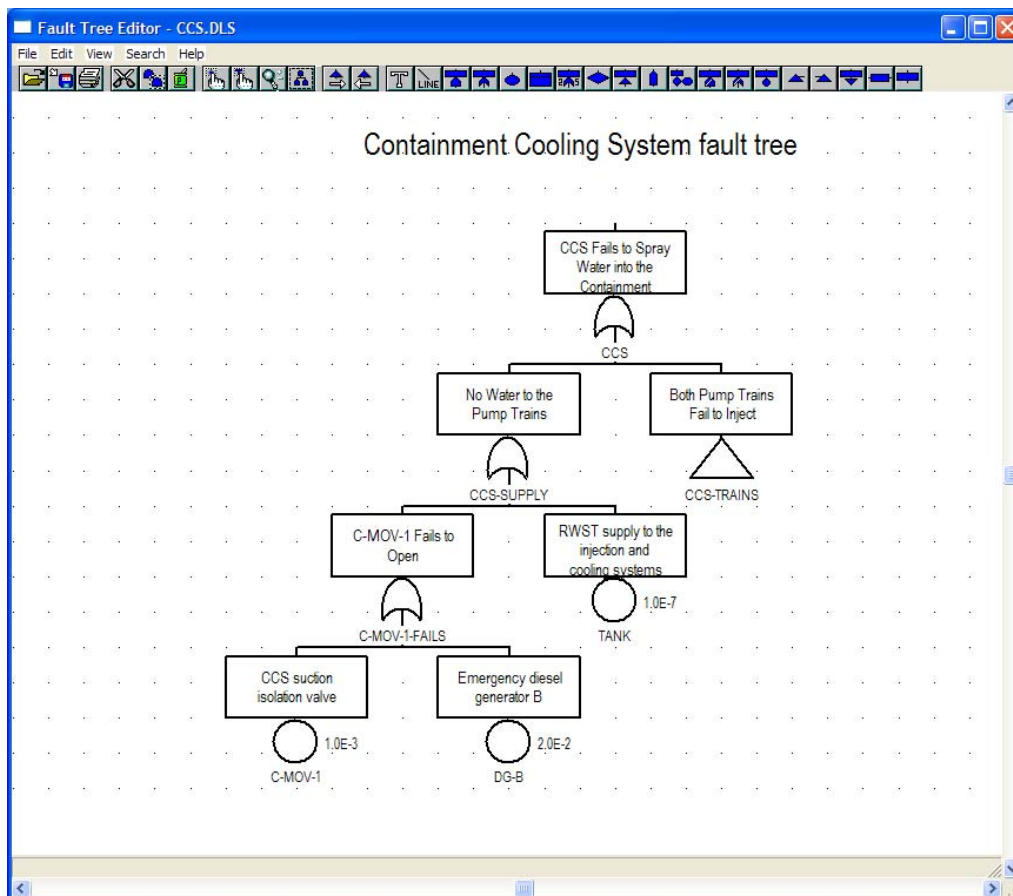
The Fault Tree Editor builds models using graphical symbols for gates, basic events, and connecting lines. These objects are tied to the SAPHIRE database. Various editing functions, such as cut, paste, and copy, are provided with file functions such as save and print.

5.2   P&ID Graphics

This graphical editor constructs piping and instrumentation flow diagrams using the Piping & Instrumentation Diagram Editor. These diagrams are used only for graphical reference and are intended to represent the physical layout of a system.

5.3   Fault Tree Logic Editor

This option provides a means for editing the fault tree logic in an alphanumeric format. All gates and gate inputs contained in the selected fault tree or sub-tree will be displayed. The fault tree logic editor is a convenient tool for quickly editing fault tree logic and entering gate and basic event data.

The logic in the editor is represented in the form of a hierarchical tree, where each gate can be expanded or condensed by clicking on the gate. The user can add new tops, rename existing tops, or add or change the inputs to existing gates.
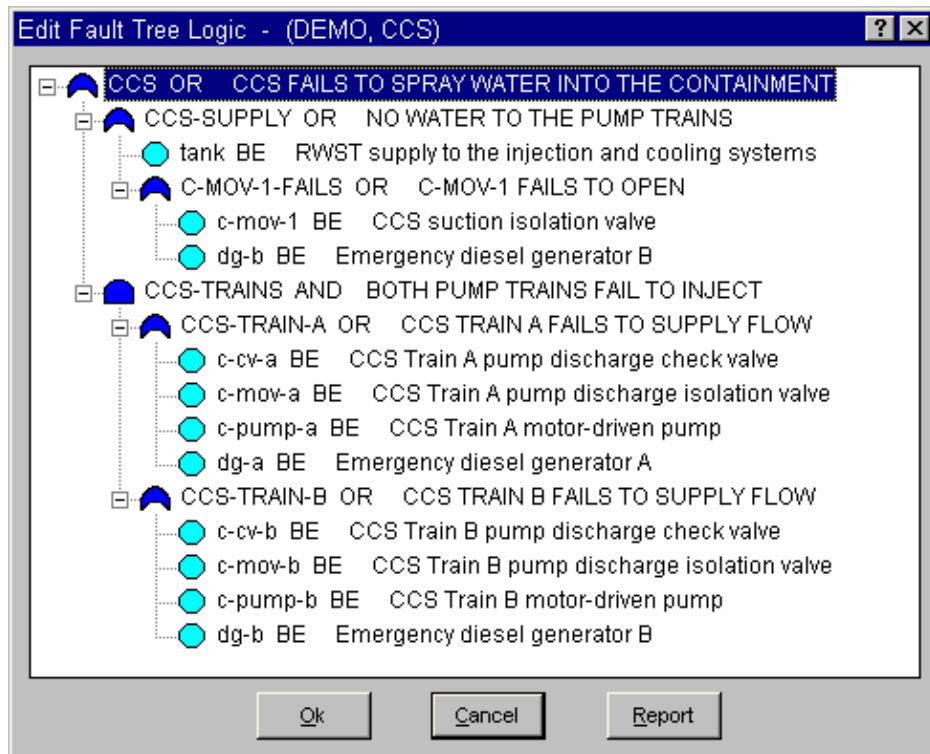
5.4   Page Fault Tree Logic

The Fault Tree Pager allows the user to organize the fault tree logic into "pages" or segments. It offers a variety of methods for making these divisions: (1) automatically divide a large fault tree logic file into a set of equivalent logic files, with each file fitting more conveniently on a page ("Auto Page"); (2) break out a sub-tree at a user-specified gate ("Gate"); (3) output gates with multiple references into a separate sub-tree ("Multiple"); or (4) output multiple top gates into separate fault trees ("Tops").

5.5　Solving Fault Trees

This option generates the cut sets for selected fault tree(s) based on cut-set generation cutoff values. The user can generate cut sets for a selected fault tree, a group of fault trees, or all fault trees within the current project. Various options allow cut sets to be generated in various ways. The user can specify several cut-set generation cutoff values that will be used to determine whether a cut set is to be retained or discarded from the selected fault tree(s). A gate that will be used as the top gate of the tree can be specified. If a name is not specified, the top gate will be determined by finding the gate not referenced by any other gate. This top gate should be named the same as the fault tree or sub-tree name. The user can select a flag set to be used during processing. If a flag set is not selected, the fault tree-specific flag set, if any, will be used.

As the calculations proceed, the *Cut Set Generation Results* dialog is displayed and updated. If a single fault tree was selected, the results are displayed in this dialog after cut set generation is complete. Choose the **Ok** button to close the dialog.

The only limit on the number of cut sets that can be stored for a given fault tree is the available space on the hard drive. When processing is complete, the *Fault Trees List* dialog is updated to show the letter "c" in front of the fault tree(s) with current case cut sets.

5.6　Quantifying Fault Trees

The quantification process will requantify the cut sets using current data values (event change sets and current-case cut sets). Quantification can be applied to a selected fault tree, for a group of fault trees, or for all of the fault trees within the current project. The new value is saved with the current case cut sets for the selected fault tree.

There are four quantification methods from which to choose: the fault trees' default method, the mincut upper bound (or Minimal Cut Set Upper Bound), min/max, and rare event approximation. The default method is set in the Modify Fault Tree option (see Section 9.3). The mincut upper bound technique approximates the probability of the union of the minimal cut sets for the fault trees. The min/max technique uses the "exact" calculation for the union of cut sets via the inclusion-exclusion rule. The rare event technique requantifies by adding together the probabilities for the cut sets of a top event.

If the selected quantification method differs from the default method of the fault tree, a warning box with the message, "The chosen quantification method was different from the Fault Tree default," is displayed. Choose the **Ok** button to continue processing.

During processing, the *Quantification Results* dialog is displayed. If an error occurs, the message, "Error quantifying cut sets," will be displayed in the message bar. If a single fault tree was selected, upon completion of the quantification process, the results are displayed in the *Quantification Results* dialog. Choose **Ok** to close the dialog.

5.7　Uncertainty Analysis

The uncertainty analysis function estimates the variability (due to the uncertainties in the basic event probabilities) of either a fault tree top event probability or an event tree sequence frequency. To do this, SAPHIRE offers two sampling techniques: Monte Carlo simulation and Latin Hypercube simulation. In addition, the user may perform a seismic analysis to integrate the seismic fragility curve with the site hazard curve using the uncertainty analysis option.

In a traditional uncertainty analysis, SAPHIRE samples the user-specified distributions for each basic event in a group of cut sets, then quantifies these cut sets using the sample values. For a seismic

uncertainty, however, SAPHIRE performs the analysis at each G-level in the site hazard curve for the current project. The seismic events are sampled at each new G-level and the quantification is performed.

An uncertainty analysis can be performed on the current case cut sets for either a single, for a group, or for all of the fault trees (sequences, or end states) within the current project.

5.7.1 Uncertainty Distributions

Thirteen types of uncertainty distributions are supported within the SAPHIRE code. The table below lists the different distributions, their identifier within SAPHIRE, and the uncertainty parameter needed by the code.

Along with the uncertainty parameter, most distributions require a second parameter. SAPHIRE requires that the mean value be specified for all distributions. The mean value is put in the database as a mean failure probability or a mean failure rate, depending on which calculation type is used.

Distribution	Identifier	Uncertainty Parameter
None (point value)	Blank	None
Lognormal	L	error factor
Normal	N	standard deviation
Beta	B	b in Beta(a, b)
Dirichlet	D	b in Beta(a, b)
Gamma	G	r in $\Gamma(r)$
Chi-squared	C	degrees of freedom
Exponential	E	none
Uniform	U	upper end point
Histograms	H	histogram name (i.e., identifier)
Maximum entropy	M	lower and upper end point
Constrained noninformative	O	none
Seismic log normal	S	Beta r, Beta u
Triangular (Version 7.x only)	T	Mode and upper end point

5.7.2 The Uncertainty Analysis Process

The following describes the uncertainty analysis process SAPHIRE performs after the user enters values in the *Uncertainty Calculation Values* dialog.

Once the number of samples has been accepted and a seed obtained from the system clock (if necessary), checks run to ensure the events with the same correlation classes have consistent failure data, uncertainty data, and distribution types. An error message will display if any events have an inconsistency, and the uncertainty analysis process will terminate so that the inconsistent values can be corrected.

If an error occurs during the uncertainty analysis process, the process terminates and a message box displays information about the specific error.

If all events successfully pass the correlation class checks, the distribution parameters for the events are checked to ensure they are valid. Error messages will display any invalid parameters, and the process will terminate so the distribution parameters can be corrected.

After both checks are met, SAPHIRE will calculate a point estimate for the selected fault tree (or sequence or end state). At this point, the samples for each event are generated using the selected sampling technique, either Monte Carlo Sampling technique or the Latin Hypercube Sampling technique. The uncertainty analysis function provides 12 distribution types for both sampling techniques. See Section 5.7.1 for information on the distribution types.

During processing, the *Uncertainty Results* dialog will display and update as the samples are generated. When the requested number of samples has been generated, statistical information is calculated using the generated samples. A sample mean, median, and standard deviation is calculated for the selected fault tree (or sequence or end state). Coefficients of skewness and kurtosis, and quantile values are also calculated and saved in the database for the selected fault tree (or sequence or end state).

Upon completion of these calculations, the following values are displayed in the *Uncertainty Results* dialog: name, random seed used, number of samples generated in this process, total number of events and cut sets being processed, point estimate, mean, median, 5th and 95th percentile values, minimum and maximum generated sample values, standard deviation, skewness and kurtosis, and time involved to perform the analysis.

If only one fault tree (or sequence or end state) was selected (highlighted) for the uncertainty analysis process, then the *Uncertainty Results* dialog must be closed. However, if more than one is being processed, the *Uncertainty Results* dialog will be displayed for each, and when all of the selected fault trees (or sequences or end states) have been processed, the *Uncertainty Results* dialog will automatically close.

5.8   Cut-Set Analysis

Cut-set analysis includes updating, pruning, editing, or performing operations on existing cut sets.

5.8.1   Updating Cut Sets

This option will update the current-case cut sets for a selected fault tree (or sequence or end state) based on cut-set generation cutoff values. The user can specify several cut-set generation cutoff values for use to determine whether a cut set will be retained or discarded. The user can also choose either the base-case cut sets or the current-case cut sets as the starting set of cut sets to be updated. The updated version of the cut sets are saved as the new current-case cut sets.

Current-case cut sets can be updated for a selected fault tree (or sequence or end state), a group of fault trees (or sequences or end states), or all of the fault trees (or sequences or end states) within the current project.

As the calculations proceed, the *Cut Set Generation Results* dialog is displayed and updated. If a single fault tree (or sequence or end state) was selected, the results are displayed in this dialog after cut-set generation is complete. Choose the **Ok** button to close the dialog.

Once the cut sets are updated, they are automatically quantified.

5.8.2   Pruning Cut Sets

This option (available in Version 6.x) allows the user to eliminate cut sets from a selected fault tree or sequence that contains events that conflict in some way with one another. This may be useful in location analysis or in eliminating technical specification violations. Note that the availability of "recovery rules" has alleviated the need for this option, and has been removed in version 7.x of SAPHIRE.

5.8.3 Recovering Cut Sets

The "recovery" option post-processes cut sets by modifying them according to a pre-defined set of rules. The user constructs recovery rules to be applied to each cut set which then define the action to be taken. The rules are constructed using the same editor that is used for event tree rules definition. This editor is a macro-based editor, which means that the user can define macros (or variables) that can be used as the search criteria tests.

The rules may be developed for a particular fault tree or all fault trees.

Item	Menu	Sub-menu option v. 6.x	Rule Level v. 7.x
Particular fault tree	Fault Trees	Edit Rules	Fault tree
All fault trees	Fault Trees	Project Rules	Project

Once the recovery rules are defined, they need to be applied. Rules can be applied by one of two methods: Batch or Interactive. The Batch Apply option applies all rules currently defined to the highlighted or marked fault trees. The Interactive option allows the user to view the quantified cut sets for a fault tree (or sequence) and interactively define and apply recovery rules before saving the recovered cut sets. The Interactive option has been removed from version 7.x.

The application of recovery rules can potentially require numerous comparisons of rules to cut sets, resulting in a long analysis time. Consequently, when the rules are applied in SAPHIRE for the entire project, the analyst is presented with the option to apply rules to cut sets down to a specified truncation level. By setting a truncation value, only those cut sets with a "high" probability will have rules applied.

5.8.4 Recovery Rules Editor

This option allows the user to create or modify rules that affect existing cut sets in a "post-processing" fashion. The rule-based editor is available for both fault tree and sequence cut sets. The Recovery Rules editor provides a means to develop logic rules. These rules allow for probabilistic risk assessment techniques such as automated inclusion of recovery events, inclusion of common-cause failure cut sets, or elimination of mutually-exclusive events (e.g., restricted or impossible combinations of events). In version 7.x, two types of rules are available: Basic and Advanced. Advanced rules are designed for users with programming skills that require the flexibility and power of using a structured programming language to program complex rules into SAPHIRE. Advanced rules are applied using the MODULA-2 programming language. Basic rules are available in both SAPHIRE 6.x and 7.x and follow a typical "if… then" type of structure.

Basic rules are entered in a free-form text editor within SAPHIRE. They follow a format similar to the structure that is found in traditional programming languages (e.g., BASIC or PASCAL). As such, the ability exists to define "macros" and "if...then" type of structures. After creating or editing rules, SAPHIRE compiles the rules to check their validity. Alternatively, rules can be exported from SAPHIRE or entered in any word processor or text editor (that can output ASCII files) and then loaded directly into the SAPHIRE database.

The recovery rules can be applied to cut sets for a particular fault tree or all fault trees in the project. Sequence recovery rules can be applied to cut sets for a particular sequence, a single event tree, or all sequences in the project.

5.8.5   Editing Cut Sets

The *Cut Set Editor* allows the user to edit the fault tree (or sequence or end state) cut sets. SAPHIRE provides space for two sets of cut sets in the database: base case and current case. Whenever SAPHIRE generates cut sets, they are stored in the current case location. The user may save a set of cut sets to the base case location by performing a Base Case Update. The user can choose to edit either base case or current case cut sets; however, **the results are always stored in the current case location**.

Using the cut set editor, the user can insert new cut sets, delete cut sets, or add or modify basic events in the cut sets. Any event name entered during cut set editing may be preceded by a "/" to indicate that it is to be treated as a complemented event. The probability of a complemented event is one minus the failure probability.

5.9   Displaying Fault Tree Results

This option displays the results of fault tree analysis functions. The cut sets and uncertainty analysis results can be displayed, and importance measure analyses can be performed.

5.9.1   Displaying Cut Sets

This option displays the current case cut sets. Two lists of cut sets are provided for fault trees (or sequences or end states). The first list, referred to as the qualified cut sets, contains those cut sets specified by the user either via the **Remove** and/or **Slice** options. The second list, referred to as the unqualified cut sets, contains those cut sets not included in the qualified cut set list. Initially, all cut sets are qualified.

When the *Selected Cut Sets* dialog is invoked, the fault tree (or sequence or end state) cut sets are shown. Displayed with each cut set is its associated percent of contribution to the minimal cut set upper bound, its probability (or, perhaps, frequency), and the event names that make up the cut sets. Also displayed are the minimal cut set upper bound for the total, the number of cut sets that make up the total, the current slice minimal cut set upper bound, the percentage that the slice contributes to the total, and the number of cut sets in the slice. From this dialog, various functions are provided as follows: View selected groups of cut sets, display the representation of the logic that generated the selected cut sets, save the cut sets to an end state, perform importance studies on selected cut sets, and produce various reports that display the cut sets.

In addition to the standard risk importance measures, SAPHIRE provides extended risk importance measures. The extended measures provided accomplish the following:

*   Prioritize the risk contributors for their risk importance and determine the cumulative risk contribution,

*   Allow risk contributions to be explicitly broken down into their probability and impact factors which allows robust, low-risk importance contributors with both low probability of failure and low-risk impact to be directly identified,

*   Obtain risk contribution and risk importance for sets of contributors to allow individual items within the set to be prioritized for resource allocation,

*   Determine the relative risk importances of the individual elements within a set with consideration for the interaction of the contributors,

*   Identify risk sensitivity for combinations of contributors when multiple contributors are changed,

*   Identify sets of contributors that have the lowest risk impact when the contributors are all relaxed or all changed by a given value.

5.9.2   Cut Set Comparison

This option displays both the base case and current case cut sets for the selected fault tree (or sequences or end states). The source (either base or current) of the cut sets is indicated by both color and a symbol. The color red and the "-" symbol in the first column indicate a base case cut set. The color black and a blank (" ") in the first column indicate that the cut set is both a current case and base case cut set. The color green and the "+" symbol in the first column indicate a current case cut set. The results can be sorted by either source (i.e., base, current, both) or probability.

5.9.3   Importance Measures

Importance measures provide "reliability-worth" information about basic events appearing in the cut sets for a fault tree or accident sequence. The "importance" for a basic event is essentially the event's contribution to the overall top event probability (for fault trees) or sequence frequency (for event trees). The measure of contribution can take on many forms depending on what concern the analyst has for a particular basic event. Consequently, SAPHIRE calculates seven different basic event importance measures. These are the Fussell-Vesely importance, risk reduction ratio, risk increase ratio, Birnbaum (or first derivative) importance, risk reduction difference, risk increase difference, and the uncertainty importance.

The ratio importance measures are dimensionless and consider only relative changes. The difference definitions account for the actual risk levels that exist and are more appropriate when actual risk levels are of concern, such as comparisons or prioritizations across different plants. For purely relative evaluations, such as prioritizations within a plant, the ratios sometimes give more graphic results.

These importance measures are calculated for each basic event for the respective fault tree or accident sequence. Once the results have been calculated, they can be displayed and then partitioned as a subset of the original cut sets.

5.9.4   Uncertainty Results

This option displays the results of the uncertainty analysis performed (See Section 5.7, Uncertainty Analysis). The results displayed are the distribution and probability bounds for both base and current results. These values were calculated using either the Latin Hypercube or the Monte Carlo simulation technique. In addition, the uncertainty quantile (or probability) values associated with either the current or the base case data can be displayed. These quantile values can be used to construct a cumulative distribution plot of the analysis uncertainty results.

5.10 Viewing Fault Tree Data

The data displayed from this option originates from the fault tree analyses previously performed. One or more fault trees can be viewed using this option. The *Fault Tree* column on this dialog contains the primary names of the fault trees. Other columns' titles and data vary based on the current "View" as follows:

- Description. Includes the full fault tree description. (60 character maximum)

- Current Vs. Base. Includes the current case cut set probability, base case cut set probability, the probability difference (current - base), ratio, current case cut set count, base case cut set count, the cut set count difference (current - base).

- Base Case. Includes base case values for cut set probability, mean, 5th percentile, median, 95th percentile, probability truncation, and size truncation used when solving.

- Current Case. Includes current case values for cut set probability, mean, 5th percentile, median, 95th percentile, probability truncation, and size truncation used when solving.

- Base Case Uncertainty. Includes base case values for cut set probability, mean, 5th percentile, median, 95th percentile, minimum, maximum, standard deviation, skewness, kurtosis, sample size, and random number seed.

- Current Case Uncertainty. Includes current case values for cut set probability, mean, 5th percentile, median, 95th percentile, minimum, maximum, standard deviation, skewness, kurtosis, sample size, and random number seed.

6.   EVENT TREES

The Event Tree Analysis option allows the user to graphically build and edit event trees and generate sequence logic. The ability to define linkage and partition rules is also provided.

When the Event Tree toolbar button or the Event Tree menu option is selected, the *Event Tree List* dialog is displayed. This dialog shows all the front-line event trees in the current project. A "+" in the first column before the event tree name indicates event tree linkage rules exist for that event tree. A "+" in the second column before the event tree name indicates partition rules exist for that event tree. A "-" in either of these columns indicates that the rules exist but are not compiled.

From this dialog the user can add a new event tree record to the database or select existing event tree(s) using the mouse or the Mask feature. The means is provided to edit the event tree graphical representation; edit linkage and partition rules; link the event trees (i.e., generate sequence logic); or edit the event tree end states.

6.1   Event Tree Graphics

The Event Tree Editor allows the user to construct or edit an event tree diagram. An event tree is a model of relationships of failed or successful fault trees or tops. An event tree diagram is a graphical representation or picture of that model.

The diagram can be built from scratch or from an existing file to generate or modify logic. The initiating event, basic events, sequences, and end states are tied to the SAPHIRE database. They are automatically added to the database as the event tree is being constructed. Various editing functions such as cut, paste, and copy along with file functions such as save and print are provided.
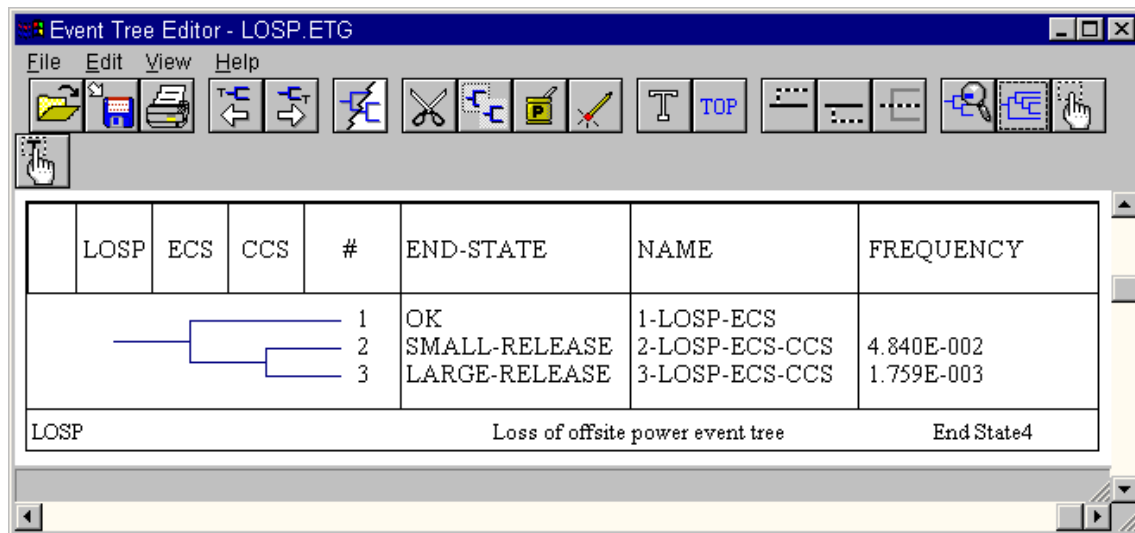
Note that an up branch indicates the success of the top event while a down branch indicates a failure of the top branch.

6.2   Linkage Rules Editor

This option allows the user to specify linkage rules for an event tree. A linkage rule is a special case, an exception, or substitution to the normal sequence generation. Typically, these rules are used to replace default fault trees (i.e., top events) with substituted fault trees (or "split-fractions") based on logical

conditions that are specified in the rules. During sequence logic generation, the logic for a sequence is created as specified in the event tree logic, and then the linkage rules are applied.

For example, event tree A contains a sequence named SEQ-1. According to the strict logic of the event tree, the fault trees that make up SEQ-1's logic are SYS-1, SYS-2, SYS-3, and SYS-4. A linkage rule would enable replacement of all occurrences of SYS-4 in any sequence in the event tree with SYS-5. Alternatively, the user may only want to replace SYS-4 with SYS-5 if and only if SYS-2 and SYS-3 are also present in the sequence logic.

The Linkage Rule Editor is a free-format line editor that provides an if-then-else logic structure. By default, the editor assumes that a name is the name of a fault tree, if one exists. If a fault tree by the name does not exist then the editor assumes that the name is the name of a macro. If the user desires to change the default assumptions, then they can "cast" the names to something else by enclosing the name in parenthesis with a cast name preceding it. There is no limit to the number of rules the user may input to the editor. Each rule is processed in sequential order.

6.2.1   Editor Functions

The editor contains file management functions such as open, save, save as, print, import, etc. Editing functions include cut, copy, paste, delete, undo, etc. Text format operations include indent, capitalization, etc. Search functions include items such as find, replace, set a marker, and go to a marker. Preference attributes can be set including items such as undo levels, select font, shift size, and show altered lines. Also, window functions can be set such as tile, cascade, and arrange icons. Finally, list options provide the ability to insert database items such as basic events, initiators, fault trees, and flags.

6.3   Partition Rules Editor

This editor allows definition of a set of rules that map the cut sets for a sequence to an end state. This is a free format, macro-based line editor that provides an if-then-else logic structure. If rules have already been defined, they will be displayed in the edit window.

By default, the editor assumes that a name is the name of a basic event, if one exists in the database. If a basic event by the name does not exist then the editor assumes that the name is the name of a macro. To change the default assumptions, the names can be "cast" to something else by enclosing the name in parenthesis with a cast name preceding it.

There is no limit to the number of rules that may be input to the editor. Each rule is processed in sequential order.

This editor operates in the same for partition rules as it does for linkage rules. See Section 6.2.1, Editor Functions, for details.

This option generates sequence logic using the event tree graphical files. During sequence generation, the sequence logic is created as specified in the event tree logic, and then the linkage rules, if any, are applied.

6.4   Editing Event Tree End States

Before sequence logic generation, a number of options can be set. A report can optionally be generated and sent to the designated output device. The report would contain the listing of the sequences' names, the fault trees that make up the sequences, any substitutions of fault trees (based on the linkage rules), and any transfers to other event trees.

The number of levels to be processed can be designated. A level refers to a transfer to a sub-tree. The default value of 99 will generate sequences whose logic will contain all tops for all the sub-trees. If the level is set to less than 99, the generated sequences will contain only tops from sub-trees less than the specified level.

If the tops are independent, the sequence logic can be truncated on probability. Each top is treated as a basic event with its probability assumed to be the value of its split fraction. As the sequence logic is being created and each top is appended to the logic, the probability for the sequence is checked to ensure that its probability is greater than the truncation value specified. All sequences with probabilities less than the truncation probability are discarded and the logic is not followed.

As a option, sequence "cut sets" can be generated when the sequence logic is generated. Each top, whether failed or successful, will be treated as a basic event and placed in a single cut set for the generated sequence. This technique is used to replace the "large event tree" methodology.

The user can optionally specify a mutually exclusive top name where the top will be added to the sequence logic. This top will be displayed in the logic as a complemented fault tree and will be treated accordingly when the sequence is solved. The purpose is to provide a method of removing illegal combinations of events. Note that this "mutually exclusive top" option has been superseded by the incorporation of recovery rules and most likely will be removed in future versions of SAPHIRE.

6.5   Editing Event Tree End States

This option allows the user to view or modify event tree sequence information after an event tree has been created. Any changes made to sequence information or end state names using this option will automatically be updated throughout the database, including the event tree graphical (.ETG) file (i.e., the corresponding drawing will be updated).

7.   ACCIDENT SEQUENCES

When developing a risk assessment code, accident sequence analysis must also be considered. Accident sequences are defined in SAPHIRE by developing event trees (see Section 6, Event Trees). Once an event tree has been developed, the sequence logic from the event tree can be generated. The sequence logic is the list of systems that succeed or fail during this accident sequence. These system failures and successes are top events of fault trees. This logic is used by SAPHIRE to generate the cut sets for the sequence.

The Sequence Analysis functions provide the means to recalculate sequence values after event failure data and/or cut sets have been modified. When the Sequence toolbar button or the Sequence menu option is selected, the *Sequences* dialog is displayed. This dialog shows all sequences in the current project, ordered by event tree name and sequence name. A "b" in front of the event tree name indicates base case cut sets exist, a "c" indicates current case cut sets exist. Uncertainty analyses can be performed and importance information about basic events can be analyzed. In addition, results can be displayed and reported in various formats.

7.1   Solving Sequences

This option generates the cut sets for selected sequence(s) based on cut set generation cutoff values. The user can specify several cut set generation cutoff values that will be used to determine if a cut set is to be retained or discarded from the selected sequence(s). Cut sets for a selected sequence, a group of sequences, or all sequences within the current project can be generated. Changes to the flags for each sequence can be specified without having to regenerate the failure data each time. The fault tree logic is pruned before it is solved, dependent on these flags. In addition, the user can choose to solve using fault tree logic or previously generated fault tree cut sets.

As the calculations proceed, the *Cut Set Generation Results* dialog is displayed and updated. If a single sequence was selected, the results are displayed in this dialog after cut set generation is complete. Choose the **Ok** button to close the dialog.

There is no limit on the number of cut sets that can be stored for a given sequence. When processing is complete, the *Sequences* dialog is updated to show the letter "c" in front of the sequence(s) with current case cut sets.

7.2   Quantifying Sequences

The quantification process will requantify the cut sets using current data values (event change sets and current case cut sets) for a selected sequence, for a group of sequences, or for all of the sequences within the current project. The new value is saved with the current case cut sets for the selected sequence.

There are five quantification methods from which to choose: the sequence default method, the mincut upper bound (or Minimal Cut Set Upper Bound), min/max, rare event approximation, or split-fraction. The default method is set in the Modify | Event Tree | Sequences option (see Section 9.2). The mincut upper bound technique approximates the probability of the union of the minimal cut sets for the fault trees. The min/max technique uses the "exact" calculation for the union of cut sets via the inclusion-exclusion rule. The rare event technique requantifies by adding together the probabilities for the cut sets of a top event. The split-fraction method quantifies the sequence cut sets using the min cut upper bound values that have been calculated for each successful or failed fault tree which make up this sequence. This option does not generate cut sets, but provides a quick approximation of the actual result. The actual results may be obtained by generating cut sets and quantifying these cut sets.

If the selected quantification method differs from the sequence(s) default, a warning box is displayed with the message, "The chosen quantification method was different from the Sequence default." Choose the **Ok** button to continue processing.

During processing, the *Quantification Results* dialog is displayed. If an error occurs, the message, "Error quantifying cut sets" will be displayed in the message bar. If a single sequence was selected, the results are displayed in the *Quantification Results* dialog upon completion of the quantification process. Choose **Ok** to close the dialog.

7.3   Sequence Uncertainty Analysis

This option operates the same for sequences as it does for fault trees. Rather than repeat the information, see Section 5.7.

7.4   Sequence Cut Set Analysis

This option provides the ability to perform a variety of analysis functions on the sequence cut sets such as updating, pruning, editing, partitioning, or recovering existing cut sets.

7.4.1   Updating Cut Sets

This option operates the same for sequences as it does for fault trees. Rather than repeat the information, see Section 5.8.1.

7.4.2   Pruning Cut Sets

This option operates the same for sequences as it does for fault trees. Rather than repeat the information, see Section 5.8.2 Note that the availability of sequence "recovery rules" has alleviated the need for this option, and has been removed in version 7.x of SAPHIRE.

7.4.3   Recovering Cut Sets

This option operates the same for sequences as it does for fault trees, except that for sequences event tree rules can also be applied. Rather than repeat the information, see Section 5.8.3. For operation of the Recover Rules Editor, see Section 5.8.4.

The rules may be developed for a particular sequence, a single event tree, or all sequences.

Item	Menu	Sub-menu option v. 6.x	Rule Level v. 7.x
Particular sequence	Sequences	Edit Rules	Sequence
Single event tree	Sequences	Edit Event Tree	Event Tree
All sequences	Sequences	Edit Project	Project

7.4.4   Editing Cut Sets

This option operates the same for sequences as it does for fault trees. Rather than repeat the information, see Section 5.8.5.

7.4.5   Partitioning Sequence Cut Sets

This option allows the user to define the end states to be associated with each cut set in an accident sequence. A set of rules that map the cut sets for a sequence to a particular end state can be defined. The rules are defined using the *Partition Rules Editor* (see Section 6.3 Partition Rules Editor for details).

Once the partition rules have been defined, they need to be applied. Rules can be applied by one of two methods: Batch or Interactive. The Apply Batch option apply all rules currently defined to the highlighted sequences. The Interactive option allows the user to interactively define and apply partition rules before saving the cut sets. Rules are sequentially applied; that is, project rules, followed by event tree rules, and finally sequence rules. If rules conflict, the latest applied rule will take precedence over earlier ones.

Sequence end states created by the partitioning rules can be cleared. By selecting the Reset End States function, the sequence end state cross-reference map is set to blank. Note cut sets will still appear in the end state list until the end state is removed. By clearing the cross-reference map, the end state can be removed, if desired, from the end states listed using the **Modify | End States** option.

7.4.6 Merge Cut Sets

The sequence "merge cut set" option allows cut sets produced when performing Level 2 analysis to be traced back to the Level 1 fault trees. Traditionally, the Level 1 plant damage state results would be rolled up into a frequency to be passed into the Level 2 event trees. The merge option allows the user to "unroll" this process in order to see what cut sets are making up the Level 1 portion of the Level 2 results.

7.5 Displaying Sequence Results

This option displays the results of sequence analysis functions. The cut sets and uncertainty analysis results can be displayed, and importance measure analyses can be performed.

7.5.1 Displaying Cut Sets

This option operates the same for sequences as it does for fault trees. Rather than repeat the information, see Section 5.9.1.

7.5.2 Cut Set Comparison

This option operates the same for sequences as it does for fault trees. Rather than repeat the information, see Section 5.9.2.

7.5.3 Importance Measures

This option operates the same for sequences as it does for fault trees. Rather than repeat the information, see Section 5.9.3.

7.5.4 Uncertainty Results

This option operates the same for sequences as it does for fault trees. Rather than repeat the information, see Section 5.9.4.

7.6 Viewing Sequence Data

The data displayed here originates from the sequence analyses previously performed. One or more sequences can be viewed with this option. The *Sequence* column contains the sequences' primary names. Other columns' titles and data vary depending on the selected "View." The "View" options are the same for sequences as they are for fault trees. Rather than repeat the information, see Section 5.10.

8.   END STATES

End State Analysis recalculates end state values after events or cut sets have been modified. Uncertainty analysis on an end state can be performed using either the Monte Carlo or Latin Hypercube simulation technique. Various functions such as updating and editing can be performed on end state cut sets. Analysis results can be displayed and reported in various formats.

When the user selects the End State toolbar button or the End State menu option, the *End State List* dialog is displayed. This dialog shows all end states defined for the current project, ordered by end state name. A "b" in front of the end state name indicates base case cut sets exist, a "c" indicates current case cut sets exist.

8.1   Gathering End States

This option *gathers* the cut sets for a selected end state based on cut set generation cutoff values. Several cut set generation cutoff values can be specified that will determine if a cut set is to be retained or discarded from the selected end state. Cut sets can be generated for a selected end state, a group of end states, or all end states within the current project. Cut sets can be gathered by the end state assigned to each sequence (via the event tree graphics) or by the end state of each cut set (assigned via end state partition rules).

As the calculations proceed, the *Cut Set Generation Results* dialog is displayed and updated. If a single end state was selected, the results are displayed in this dialog after cut set generation is complete. Choose the **Ok** button to close the dialog.

The limit on the number of cut sets that can be stored for a given end state is constrained by the available hard disk space. When processing is complete, the *End State List* dialog is updated to show the letter "c" in front of the end state(s) with current case cut sets.

8.2   Quantifying End States

The quantification process will calculate a new minimum cut set upper bound for the end state cut sets using the current data values (event change sets and current case cut sets). The new value is saved with the current case cut sets for the selected end state.

There are three quantification methods from which to choose: the end states' default method, the mincut upper bound (or Minimal Cut Set Upper Bound), or rare event approximation. The default method is set in the Modify | End State option (see Section 9.4). The mincut upper bound technique approximates the probability of the union of the minimal cut sets for the end states. The rare event technique requantifies by adding together the probabilities for the cut sets of a top event.

If the selected quantification method differs from the end state's default, a warning dialog is displayed with the message, "The chosen quantification method was different from the End State default." Choose the **Ok** button to continue processing.

During processing, the *Quantification Results* dialog is displayed. If an error occurs, the message, "Error quantifying cut sets" will be displayed in the message bar. If a single end state was selected, upon completion of the quantification process, the results are displayed in the *Quantification Results* dialog. Choose **Ok** to close the dialog.

8.3 End State Uncertainty Analysis

This option operates the same for end states as it does for fault trees. Rather than repeat the information, see Section 5.7.

8.4 End State Cut Set Analysis

End state cut set analysis functions consist of updating or editing cut sets.

8.4.1 Updating Cut Sets

This option operates the same for end states as it does for fault trees. Rather than repeat the information, see Section 5.8.1.

8.4.2 Editing Cut Sets

This option operates the same for end states as it does for fault trees. Rather than repeat the information, see Section 5.8.5.

8.5 Displaying End State Results

This option displays the results of end state analysis functions. Cut sets and uncertainty analysis results can be displayed, and importance measure analyses can be performed.

8.5.1 Displaying Cut Sets

This option operates the same for end states as it does for fault trees. Rather than repeat the information, see Section 5.9.1.

8.5.2 Cut Set Comparison

This option operates the same for end states as it does for fault trees. Rather than repeat the information, see Section 5.9.2.

8.5.3 Importance Measures

This option operates the same for end states as it does for fault trees. Rather than repeat the information, see Section 5.9.3.

8.5.4 Uncertainty Results

This option operates the same for end states as it does for fault trees. Rather than repeat the information, see Section 5.9.4.

8.6 Viewing End State Data

The data displayed here originates from the end state analyses previously performed. One or more end states can be viewed with this option. The *End State* column contains the end states' primary names. Other columns' titles and data vary depending on the selected "View." The "View" options are the same for end states as they are for fault trees. Rather than repeat the information, see Section 5.10.

9.   MODIFYING DATABASE TABLES

This option allows modifying the various SAPHIRE tables: project, fault tree, event tree, sequence, end state, basic event, attribute, gate, histogram, piping and instrumentation diagram, change set, and flag set. The records in the database can be added, modified, copied, and deleted.

9.1   Modifying Project Data

This option modifies data for the current project. Data fields include name, description, location, company, type, design, vendor, model type, default locale, site hazard curves, operational date, qualification date, and mission time.

9.2   Modifying Event Tree Data

This option allows adding, modifying, copying, or deleting event tree data records or associated text. In addition, selected event tree base-case data can be updated or selected event tree current-case data can be cleared. You can add, modify, copy, or delete sequence data records associated with the selected event tree. Data fields include name, description, and initiator.

9.2.1   Base-Case Update

This option overwrites all base-case (original) data with the current case data for the selected analysis type. If you chooses to update the base data, four operations will occur: (1) the base-case cut sets will be overwritten with the current case cut sets; (2) the base case event tree uncertainty results will be overwritten with the current case uncertainty results; (3) the base case uncertainty quantile values will be overwritten with the quantile values from the current case; and (4) the base-case minimum cut set upper bound will be overwritten with the current case minimum cut set upper bound. Caution: *the original event tree base case results will be overwritten if this option is executed!*

9.2.2   Clear Current Case

This option deletes all current case information for the specified analysis type. All SAPHIRE calculations use the data stored in the current case for sensitivity and event analysis, and cut set generation results are stored in the current case.

9.2.3   Modifying Sequence Data

This option allows adding, modifying, copying, or deleting data records or text for the sequences associated with an event tree. In addition, you can update selected sequence base-case data or clear selected sequence current-case data. See Sections 9.2.1 and 9.2.2. Data fields include name, description, end state, flags name, and quantification method.

9.3   Modifying Fault Tree Data

This option allows adding, modifying, or deleting fault tree data records. Data fields include name, description, system code, sub-tree, flags name, probability cut off value, and quantification method. When modifying an existing fault tree record, current-case mincut upper bound values for each of the 16 analysis types can be entered.

9.4   Modifying End State Data

This option allows adding, modifying, copying, or deleting end state data records or associated text. In addition, you can update selected end state base-case data or clear selected end state current-case data. See Section 9.2.1 and Section 9.2.2. Unused end state records can be deleted from the database. Data fields include name, description, and quantification method.

9.5   Modifying Basic Event Data

This option allows adding, modifying, or deleting basic event data records. Deletion of all unused basic event records is also provided.

The cross-reference map of the highlighted event can be viewed and reported here. This cross-reference map shows the fault tree logic, change sets, and cut sets in which the basic event is used. For each fault tree, sequence, and/or end state in which the basic event is used, base-case and/or current-case cut sets will be listed for each analysis type.
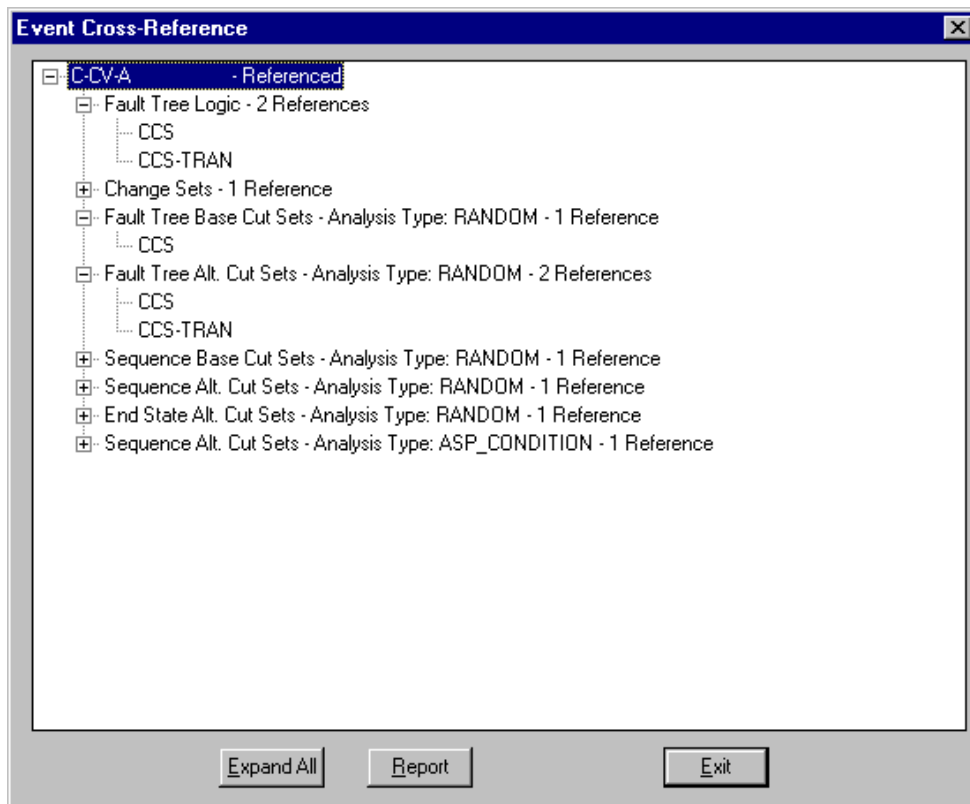
When an event from the *Edit Events* list is selected, the *Modify Event* property sheet is displayed. Each tab on the property sheet contains fields grouped by related properties or functions. These fields are listed by property sheet tab:

9.5.1   Event Tab

Fields on this tab include primary name, alternate name, description, calculation type, mean failure probability, lambda (failure rate), tau (the mean time to repair or test interval, depending on the calculation type), mission time, uncertainty distribution type, distribution type value 1, distribution type value 2, and correlation class.

9.5.2   Attributes Tab

Fields on this tab include component identifier, system, train, type, failure mode and location attributes, category, graphical shape, special use flags, and analysis type susceptibilities.

9.5.3   Process Flag Tab

The Process Flag field specifies whether certain processes should take special note of the selected event. On this tab, you can select from the following process flags (only one can be selected for each event):

Sequence Top Flags:

Default (or Blank)	The transfer associated with this event is expanded for failure references. For success references, the transfer is also expanded; however, the success cut sets generated are removed from the resulting failure cut sets using cut set matching.
"I"	Use fault tree logic (if top event fails), or use the complement of the fault tree logic (if top event succeeds). That is, if the top event is a failure, SAPHIRE will expand the fault tree and solve; if the event succeeds, SAPHIRE will complement the fault tree logic and solve it.
"W"	Use fault tree logic (if top event fails), or 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 use the complement of the developed event (i.e., just a probability associated with the event) for the fault tree.
"X"	Use developed event (if event fails), or use cut set matching to eliminate cut sets (if event succeeds). That is, an "X" tells SAPHIRE that the top event is to be used (as a probability) for failure references, but success references are to be treated the same as if the flag were blank.
"Y"	Use developed event (if event fails), or 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 references.

9.5.3.1   Sequence and Fault Tree Logic Flags

Default (or Blank)	Use fault tree logic
"X"	Always use developed event. That is, do not expand the fault tree or transfer. Instead, simply use the event probability.

9.5.3.1.1   General Flags

Sensitivity Analysis	SAPHIRE will map a results frequency or probability plot. A sensitivity analysis indicates how sensitive the frequency or probability is in relation to an event.
Zone "flagged" event	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.

9.5.4   Template Tab

A template event is a basic event that can be used to define all or part of another basic event. Template events are generally used for "generic" types of data wherein a probability value (or failure rate) is to be identical across multiple basic events.

On this tab, the template event whose attributes will be used can be identified. Then, the attributes, failure data, uncertainty data, and miscellaneous fields that will be copied from the selected template event can be chosen. By default, all of the values of the template will be copied to the basic event using the template. A check mark in the box indicates that the template event's value will be used.

9.5.5   Transformations Tab

A transformation is a replacement or addition of basic event(s) inside the fault tree logic. During the transformation process, an event is replaced by one or more events, as specified by the transformation. A transformation will also occur when a zone-flagged event is encountered.

On this tab, the transformation basic events are designated. The list on the left displays all basic events defined for the current project. The list on the right of the dialog displays the events that have been marked as replacements for the current event.

9.5.6   Uncertainty Tab

This tab shows a graphical plot of the uncertainty distribution defined by the failure and uncertainty values.

9.5.7   Compound Event Tab

This tab provides the means to calculate the compound event's probability. A compound event's probability is defined by other basic events' probability values. The tab is available only when the **Compound Event** radio button is selected on the *Modify Event Attributes* tab.

To calculate the compound event's probability, a SAPHIRE dynamic link library (DLL) file must be selected from the *Library* drop-down list. Once the DLL is selected, a procedure from the *Procedure* drop-down list must be chosen. After the desired procedure is selected, a list of appropriate parameters will be displayed in the *Parameter* column of *Selected Event* list. Choose an event from the *All Events* list for each parameter in the *Selected Event* list.

The *All Events* list on the left displays all basic events defined for the current project. The *Selected Event* list on the right of the dialog displays the events that have been selected as parameters for the selected DLL library procedure.

Plug-in Name	Description	Availability
PlugCCFAlpha.DLL	This function returns the Common Cause Failure probability (assuming nonstaggered testing) using the alpha-factor method. Inputs are the failure count, the associated basic events, and alpha factors.	Version 6.x and 7.x
PlugCCFStag.DLL	This function returns the CCF probability (assuming staggered testing) using the alpha-factor method. Inputs are the failure count, the associated basic events, and alpha factors.	Version 6.x and 7.x
PlugCCFMGL.DLL	This function returns the Common Cause Failure probability using the multiple Greek letter (MGL) method. Inputs are the failure count, associated basic events, and the MGL factors.	Version 6.x and 7.x
PlugUtil.DLL	This library provides a basic set of mathematical functions, including multiply, divide, add, subtract, natural log, exponent, power, sine, cosine, tangent, minimal cut set upper bound calculation, and absolute value.	Version 6.x and 7.x
PlugCapacityload.DLL	This plug-in performs a 'capacity-load' calculation. The capacity and load are basic events representing physical conditions like pressure, temperature, material strength, etc.. Using these events, the plug-in finds when the load is larger than the capacity: P(capacity < load) = failure probability	Version 7.x

PlugTimeSeries.DLL	Calculates the probability of that the sum of a sequential series of timed events exceeds the given mission time. Inputs are Mission Time and up to 19 other time duration-related events.	Versions 6.x and 7.x
Plug_LoopRec.DLL	This function sums the frequency (lambda) for the five loss-of-offsite-power categories: plant centered, grid related, switchyard related, severe weather, and extreme weather. The function can also provide a frequency weighted average probability of not recovering from the loss of power.	Version 7.x

9.5.8   Notes Tab

This tab allows for addition of descriptive text to accompany the basic event. Text can include analyst notes and data calculations, data sources and references, assumptions, and any other relevant information that may pertain to the basic event.

9.5.9   Basic Event Cross-reference

This option displays a cross-reference map of the highlighted basic event. This cross-reference map will show the fault tree logic, change sets, and cut sets in which the basic event is used. For each fault tree, sequence, and/or end state in which the basic event is used, base and/or current case cut sets will be listed for each analysis type.

9.5.10   Remove Unused Basic Events

This option looks at all references to an event in the current project and deletes any events that are not referenced by anything.

9.6   Modifying Attribute Data

This option allows editing of records of the five attribute types in the current project: Systems, Locations, Failure Modes, Basic Event Types, and Trains. Data fields on the Modify dialog are the attribute name and description.

9.7   Modifying Analysis Type Data

This option allows modifying analysis types. SAPHIRE offers eight predefined analysis types and eight user-defined analysis types. The names of the eight predefined analysis types cannot be edited, but the description can be changed. Both the name and descriptions of the eight user-defined analysis types can be edited.

9.8   Modifying Gate Data

This option allows adding, modifying, or deleting the gate data records. Data fields include name, description, and gate type.

9.9   Modifying Histogram Data

This option allows adding, modifying, or deleting user-defined histogram records. This is a useful option for entering a user-specified distribution for a variable that cannot be expressed with one of the predefined uncertainty distribution types. Histograms can be in percentage, range, area, or hazard format.

9.10 Modifying P&ID Data

This option allows editing of Piping and Instrumentation Diagram (P&ID) graphic records. Data fields on the Modify dialog are the name and description.

9.11 Modifying Change-Set Data

This option allows adding, modifying, deleting, or copying change-set records. To edit the probability or class changes of event data in the change set, see Section 4, Generating Event Data. Data fields include name, description, date, set type, and a read-only flag.

9.11.1 Base-Case Event Update

This option overwrites the base-case data values stored in the database with the current-case data values for each basic event. After executing this operation, the original base-case data are no longer available.

To perform a base-case update, at least one change set must be marked and data generated. This option transfers the current values for the basic events marked either with a probability change or a class change in a change set to the base-case values. The existing base-case values will be lost. Any event in the change set that does not have either a class change or a probability change associated with it will maintain its existing base-case data.

9.12 Modifying Flag Set Data

This option allows adding, modifying, deleting, or copying flag set records. Flag sets are a special type of change set used to indicate modifications to particular events sequence by sequence. They are different from change sets in that they can contain only individually specified types of changes. No "Class Changes" are allowed in a flag set. Data fields include name, description, date, set type, and a read-only flag.

9.12.1 Flag Set Events

This option allows editing or adding house flag changes to the calculation type or process type. Changes to the probability of failure (e.g., change the probability from 2E-3 to 1E-1) for events in a Flag Set cannot be made. Instead, only the event changes of TRUE, FALSE, IGNORE, or process flags can be used in a SAPHIRE flag set. The changes made can be applied to specified fault trees or sequences for use during fault tree or event tree analysis. The changes applied to basic event data may be for a single event or for a group of events.

10. GENERATING REPORTS

The report option obtains information about the current project. Various reports can be generated for the following data types: project, attributes, basic events, fault trees, event trees, end states, sequences, gates, histograms, and user information. The reports include options such as summary, cross-reference, and text information. In addition, a custom report option is available for each of the data types, which allows the user to create reports on an ad-hoc basis. The dialogs and operations are similar for each of the report data types; therefore, a generic discussion is provided:

1.   Choose the Report toolbar button or the Report option from the SAPHIRE menu.

2.   From the *Reports Menu* dialog, select a specific report by choosing a radio button from each group (*Data Type, Report Type*, and *Sub Type*, if necessary).

3.   Choose the **Process** button.

4.   A dialog containing the list of records associated with the selected report type will be displayed.

5.   Select the desired records.

6.   Choose the button on the dialog to generate the report. Often, this button is labeled **Continue,** or it may be for a specific type of report, such as, **Summary**.

7.   Specify the output destination for the report (e.g., printer, file, or report viewer).

In addition to report type-specific information, each report will contain the report title and the date and time generated. Where applicable, it will also contain the currently selected project name, selected case (base or current), and page numbers.

10.1 Custom Reports

This option obtains information about the selected data type (e.g., project, fault tree, basic event) on an ad-hoc basis. When creating a custom report, the database fields of the selected data type are available for inclusion in the report. The fields of interest can be selected, headers for the fields renamed, and the report can be saved. All records of the data type are included when the custom report is generated. Multiple ad-hoc report formats can be created and saved in the SAPHIRE database.

10.2 Report Output

When reporting information about the current project, specification of the output destination for the report is required. Three options are available: Print Preview, Printer, and File. The File option saves the output to a user-named file in the desired folder. The Printer option sends the output to the Windows system default printer. The Print Preview option displays the report in the *Report Viewer* window.

The *Report Viewer* allows scrolling through the report. A temporary file is created in the current project folder containing the generated report data. Various options such as Save As, Print, and Copy are provided from the *Report Viewer* menu.

11. UTILITY FUNCTIONS

The Utility options perform routine functions required by SAPHIRE, such as defining constants, recovering the database, and MAR-D data exchange.

11.1 Defining Constants

The Define Constants option provides the means to specify general default information, cut-set option defaults, and graphical editor defaults. The define constants dialog consists of four property sheets, each containing information pertaining to defaults, grouped by related properties.

The *General* tab allows setting SAPHIRE general values such as the default analysis type, menu, and toolbar settings; default uncertainty values; and default importance measurement values. The *Cut Set* tab allows setting default values for cut set generation and analysis, and end state gathering. Default truncation, cut set solving, quantification, and transformation methods and values are set on this tab. The *Fault Tree* tab allows setting the default fault tree attributes for shapes, lines, and text, such as font, color, and alignment. The default window attributes such as background color, grid, name, text, and probability display can also be set. The *Event Tree* tab allows setting the default event tree attributes for the tops and text. In addition, window attributes such as background color and leaf height can be set.

11.2 Recovering the Database

The Recover Database option restructures the database and reindexes the data. The database can be rebuilt at anytime because the rebuild process compacts the data and generally helps the software run faster. This option will recover all key indexes and then recover the cross-references. This process may take several minutes to complete. If the database has not been damaged, this option will just restructure and optimize the database file size.

11.3 Load and Extract

Load and Extract loads probabilistic risk assessment data from the MAR-D or Set Equation Transformation System (SETS) database into SAPHIRE. In addition, SAPHIRE-formatted data can be output to the generic MAR-D format or SETS format. The Load operation converts information from the generic format found in the MAR-D database to the SAPHIRE database format. New data loaded into the database overwrites old data of the same name.

The Extract option outputs data in MAR-D file format. The extracted file is created in the current project folder. The default output file name for description, information, and attribute data is the project name plus the appropriate extension. For cut set, logic, graphic, and text information, the name of the file is generated based on the selected output option and internal counters. If a file with the generated name already exists, it will be overwritten.

11.4 Fault Tree and Event Tree Options

The ability to manage fault tree and event tree graphics files is also provided. Fault tree logic can be converted from alphanumeric format to graphical format. Also, fault tree and event tree graphics data can be loaded or extracted via MAR-D files.

12. GEM OVERVIEW

GEM is a special application for performing nuclear power plant operational event assessment using the SAPHIRE-based SPAR models and methods. GEM automates the processes used to calculate CCDP.

12.1 Using GEM

The GEM code is integrated with the SAPHIRE code and is automatically installed with SAPHIRE, but runs as a separate application. The start (choose the GEM icon instead), exit, first-time, and conversion operations are the same for GEM as for SAPHIRE. Rather than repeat the information, see to Section 2.2. In addition, the utility operations for defining application constants and recovering the database operate the same for GEM as for SAPHIRE. See Sections 11.1 and 11.2 for details.

An on-line reference guide in the form of a hypertext help system is supplied with GEM, providing step-by-step instructions for operation. Because the help system operates the same for GEM as it does for SAPHIRE, see Section 2.4 for a general discussion of the on-line help system.

12.2 Initiating Event Assessments

An initiating event assessment evaluates an operation occurrence associated with an observed initiating event. An initiating event assessment also considers possible involvement of other equipment failures, unavailabilities, or degradations that impact the ability of the plant and plant operators to properly respond to the initiating event.

ASP models contain event trees that model the plant response to a selected set of initiating events. When a precursor to be analyzed involves one of these initiating events, an initiating event assessment is performed. The analyst invokes GEM and after selecting the correct plant model, selects an initiating event assessment and gives it a name. This opens a work space for this particular initiating event assessment.

The analyst next selects the initiating event that occurred from the list provided. GEM will automatically set the selected initiating event's frequency to its preassigned, short-term, nonrecovery probability (or 1.0 if recovery is deemed not possible) and will set all other initiating event frequencies to zero. This assumes that the actual event scenario involves only the observed initiating event, and all other initiators are not applicable. Furthermore, it is assumed that the probability of another initiating event occurring while the first initiating event is still in progress is small enough to be ignored.

If the initiating event is a loss of off-site power (LOOP) initiator, GEM will display a list of the four types of LOOP (i.e., plant-centered, grid-related, severe weather-related, and extremely severe weather-related) and the analyst must pick one. Each LOOP type has a different short-term, nonrecovery probability.

Once the initiating event is selected, GEM displays a predefined change set unique to that initiating event. For all initiating events except LOOP, the change set consists only of the entire set of initiating events. The selected initiating event is in the change set because its frequency has been converted to a probability equal to its short-term, nonrecovery value, and the other initiating events are listed because their frequencies have been converted to a probability of zero.

For the LOOP initiating event, the change set again lists all the initiating events, but in addition, the change set includes a predefined list of electric power recovery basic events. For pressurized water reactors (PWRs), the change set includes the reactor coolant pump (RCP) seal loss-of-coolant accident (LOCA) basic event. These are the basic events that must be changed when going from the nominal LOOP event in the base case model to a specific LOOP type for the event assessment. These basic events

(defined using special compound events) are automatically added to the change set, and their values are automatically modified to predetermined values based on the plant characteristics, the timing required, and the type of LOOP.

The analyst can access any of the basic events listed in the change set if modifications are needed. Other basic events can also be added to the change set list as desired. This would be necessary to model any observed failures or degradations reported for the event being analyzed.

Once the change set list is complete and the desired changes are defined for the basic events, the analyst selects the Process option. GEM automatically generates a current basic event data set according to the changes defined in the change set. GEM next determines which system fault trees have been impacted by the changes and identifies the event tree accident sequences that use those fault trees. GEM solves the logic for those sequences and produces the new minimal cut sets. The minimal cut sets are quantified, the CCDP is calculated, and the results are displayed. With the exception of the CCDP calculation, GEM processes the accident sequences in exactly the same manner as SAPHIRE.

12.3 Condition Assessments

When a precursor event does not involve the actual occurrence of an initiating event, but does indicate failures or degradation of equipment that would be required to mitigate the consequences of an initiating event, a condition assessment is performed. After entering GEM and selecting the appropriate ASP model, the analyst selects the condition assessment option and gives the assessment a name. This opens up a work space for this particular condition assessment. GEM displays a blank change set and provides the ability to add events to the change set to model the observed failures and degradations reported for the event being analyzed.

Once the change set list is complete and the desired changes are defined for the basic events, the analyst selects the Process option. GEM automatically generates a current basic event data set according to the changes defined in the change set. GEM next determines which system fault trees have been impacted by the changes and identifies the event tree accident sequences that use those fault trees. The logic for those sequences is solved for the new minimal cut sets, and the minimal cut sets are then quantified. For sequences not impacted by any of the changes, the base-case Random analysis results are copied into the Condition Assessment results.

GEM then prompts the analyst for the duration of the conditions being analyzed. The duration is entered, expressed in hours.

The CCDP is calculated, and the results are displayed. With the exception of the CCDP calculation, GEM processes the accident sequences in exactly the same manner as SAPHIRE.

12.4 LERF Assessments

The GEM LERF assessment feature can evaluate PRA model specific accident sequences, from accident initiation through LERF. The "LERF" option in GEM allows a user to select an existing initiating event assessment or condition assessment; change set; or create a new assessment, but evaluate the assessment using a LERF end state gather instead of the core damage sequences. GEM has been designed to look specifically for sequences labeled "PDS-" (either on the event tree graphic or created via event tree partition rules). As such, the plant model must be designed such that only LERF sequences are assigned to end states beginning with "PDS-" for the correct LERF results to be generated. In addition to the correct end state assignments, the model must be specified as "LERF-enabled" (see Vol. 3, section 9.2.9).

13. SAPHIRE QUALITY ASSURANCE

SAPHIRE has been extensively validated in several ways. Complete details of the QA process can be found in Volume 6 of this NUREG/CR series. This section summarizes SAPHIRE's validation processes.

13.1 The Beta Version Process

Each new SAPHIRE version is tested before its release. The beta testers are analysts experienced with PRA methods and terminology and, typically, are familiar with earlier versions of SAPHIRE and IRRAS. The primary objective of the beta testing is to verify that the results produced by the new version are correct. The secondary objective is to ensure that the software is user-friendly and functional. Based on feedback from beta testers, programmatic errors and discrepancies are corrected, the user interface has its functionality improved, and new features are recommended.

13.2 Traditional Validation and Verification

A verification and validation (V&V) of SAPHIRE 5.0 in the traditional sense was performed in 1995 and documented in NUREG/CR-6116, Volume 9, *Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE) Version 5.0, Verification and Validation (V&V) Manual*, 1995. However, this V&V process was found not to be cost-effective because it uses most of the available resources for documentation rather than for actual testing.

13.3 PRA Loading and Result Comparison

Because INL is continually loading new PRAs into SAPHIRE, new SAPHIRE releases are tested extensively by (a) comparing them with PRA models and results of earlier versions and especially with PRAs loaded in SAPHIRE Version 5.0, and (b) loading new PRAs and comparing them with the original PRA models and results.

Over 30 PRAs have been (completely or partially) loaded into SAPHIRE from other software packages. The loading process requires detailed comparison of the models and results with the models and results of the original PRAs. Identified discrepancies are resolved through interaction with the individual licensees and analysts who performed the PRA. Given that different PRAs have been performed with different types of software, one can argue that SAPHIRE has been tested with an enormous number of test cases.

13.4 Automated Code Testing

Owing to the advances in computer science, an automated testing process was established during 1998 that allows test engineers to develop and maintain scripts for selected critical functions of the software. Before each release, these scripts are run against the SAPHIRE software on a number of machines with varying configurations. The test results are automatically compared and any discrepancies are resolved. The automated testing process has been used for Windows Versions 6 and 7 of SAPHIRE and is documented in Volume 6 of this NUREG/CR series.

14. REFERENCES

K. D. Russell, et al. *Systems Analysis Programs for Hands-on Reliability Evaluations (SAPHIRE) Version 6.0 - System Overview Manual*, NUREG/CR-6532, May 1999.

K. D. Russell et al., *Integrated Reliability and Risk Analysis System (IRRAS) Version 5.0, Volume 2 - Reference Manual*, NUREG/CR-6116, EGG-2716, July 1994.

K. D. Russell et al., *Verification and Validation (V&V), Volume 9 – Reference Manual*, NUREG/CR-6116, EGG-2716, July 1994.

K. D. Russell et al., *Integrated Reliability and Risk Analysis System (IRRAS) Version 4.0, Volume 1 - Reference Manual*, NUREG/CR-5813, EGG-2664, January 1992.

K. D. Russell et al., *Integrated Reliability and Risk Analysis System (IRRAS) Version 2.5 Reference Manual*, NUREG/CR-5300, EGG-2613, March 1991.

K. D. Russell, M. B. Sattison, D. M. Rasmuson, *Integrated Reliability and Risk Analysis System (IRRAS) - Version 2.0 User's Guide*, NUREG/CR-5111, EGG-2535, manuscript completed March 1989, published June 1990.

K. D. Russell, D. M. Snider, M. B. Sattison, H. D. Stewart, S.D. Matthews, K. L. Wagner, *Integrated Reliability and Risk Analysis System (IRRAS) User's Guide - Version 1.0 (DRAFT)*, NUREG/CR-4844, EGG-2495, June 1987.

U.S. NUCLEAR REGULATORY COMMISSION

BIBLIOGRAPHIC DATA SHEET

(See Instructions on the reverse)

1. REPORT NUMBER (Assigned by NRC, Add Vol., Supp., Rev., and Addendum Numbers, if any.)

NUREG/CR-6952
INL/EXT-05-00248

2. TITLE AND SUBTITLE

Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE) Vol. 1 Summary Manual

3. DATE REPORT PUBLISHED

MONTH	YEAR
September	2008

4. FIN OR GRANT NUMBER

N6203

5. AUTHOR(S)

S. T. Wood, C. L. Smith, K. J. Kvarfordt, S. T. Beck

6. TYPE OF REPORT

Technical

7. PERIOD COVERED (Inclusive Dates)

8. PERFORMING ORGANIZATION - NAME AND ADDRESS (If NRC, provide Division, Office or Region, U.S. Nuclear Regulatory Commission, and mailing address; if contractor, provide name and mailing address.)

Idaho National Laboratory
Battelle Energy Alliance
P.O. Box 1625
Idaho Falls, ID 83415-3850

9. SPONSORING ORGANIZATION - NAME AND ADDRESS (If NRC, type "Same as above"; If contractor, provide NRC Division, Office or Region, U.S. Nuclear Regulatory Commission, and mailing address.)

Division of Risk Analysis
Office of Nuclear Regulatory Research
U.S. Nuclear Regulatory Commission
Washington, DC 20555-0001

10. SUPPLEMENTARY NOTES

D. O'Neal, NRC Project Manager

11. ABSTRACT (200 words or less)

The Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE) is a software application developed for performing a complete probabilistic risk assessment (PRA) using a personal computer (PC) running the Microsoft Windows operating system. SAPHIRE is primarily funded by the U.S. Nuclear Regulatory Commission (NRC) and developed by the Idaho National Laboratory (INL). INL's primary role in this project is that of software developer and tester. However, INL also plays an important role in technology transfer by interfacing and supporting SAPHIRE users, who constitute a wide range of PRA practitioners from the NRC, national laboratories, the private sector, and foreign countries. SAPHIRE can be used to model a complex system's response to initiating events and quantify associated consequential outcome frequencies. Specifically, for nuclear power plant applications, SAPHIRE can identify important contributors to core damage (Level 1 PRA) and containment failure during a severe accident which lead to releases (Level 2 PRA). It can be used for a PRA where the reactor is at full power, low power, or at shutdown conditions.

12. KEY WORDS/DESCRIPTORS (List words or phrases that will assist researchers in locating the report.)

SAPHIRE, software, reliability, risk, safety, PRA

13. AVAILABILITY STATEMENT

Unlimited

14. SECURITY CLASSIFICATION

(This page)

Unclassified

(This report)

Unclassified

15. NUMBER OF PAGES

16. PRICE