JUL 03 1990

Integrated Reliability and Risk Analysis System (IRRAS) Version 2.0 User's Guide

Prepared by K. D. Russell, M. B. Sattison
D. M. Rasmuson

Idaho National Engineering Laboratory
EG&G Idaho, Inc.

Prepared for
U.S. Nuclear Regulatory
Commission

DO NOT MICROFILM
COVER

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, 2120 L Street, NW, Lower Level, Washington, DC 20555
2. The Superintendent of Documents, U.S. Government Printing Office, P.O. Box 37082, Washington, DC 20013-7082
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 Office of Inspection and Enforcement bulletins, circulars, information notices, inspection and investigation 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, 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 and periodical 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 Information Resources Management, Distribution Section, U.S. Nuclear Regulatory Commission, Washington, DC 20555.

Copies of industry codes and standards used in a substantive manner in the NRC regulatory process are maintained at the NRC Library, 7920 Norfolk Avenue, Bethesda, Maryland, and are available there for reference use by the public. 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

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, or any of their employees, makes any warranty, expressed or implied, or assumes any legal liability of responsibility for any third party's use, or the results of such use, of any information, apparatus, product or process disclosed in this report, or represents that its use by such third party would not infringe privately owned rights.

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

Integrated Reliability and Risk Analysis System (IRRAS) Version 2.0 User's Guide

Manuscript Completed: March 1989
Date Published: June 1990

Prepared by
K. D. Russell, M. B. Sattison/INEL
D. M. Rasmuson/NRC

Idaho National Engineering Laboratory
Managed by the U.S. Department of Energy

EG&G Idaho, Inc.
P.O. Box 1625
Idaho Falls, ID 83415

Prepared for
Division of Systems Research
Office of Nuclear Regulatory Research
U.S. Nuclear Regulatory Commission
Washington, DC 20555
NRC FIN A6399

MASTER
DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

Previous Reports in the Series

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 Integrated Reliability and Risk Analysis System (IRRAS) is a state-of-the-art, microcomputer-based probabilistic risk assessment (PRA) model development and analysis tool to address key nuclear plant safety issues. IRRAS is an integrated software tool that gives the user the ability to create and analyze fault trees and accident sequences using a microcomputer. This program provides functions that range from graphical fault tree construction to cut set generation and quantification. Also provided in the system is an integrated full-screen editor for use when interfacing with remote mainframe computer systems.

Version 1.0 of the IRRAS program was released in February of 1987. Since that time, many user comments and enhancements have been incorporated into the program providing a much more powerful and user-friendly system. This version has been designated IRRAS 2.0 and is the subject of this user's guide. Version 2.0 of IRRAS provides all of the same capabilities as Version 1.0 and adds a relational data base facility for managing the data, improved functionality, and improved algorithm performance.

SUMMARY

The Integrated Reliability and Risk Analysis System (IRRAS) is being developed at the Idaho National Engineering Laboratory (INEL) as the U.S. Nuclear Regulatory Commission's (NRC's) state-of-the-art microcomputer-based probabilistic risk assessment (PRA) model development and analysis tool to address key nuclear plant safety issues.

IRRAS is an integrated PRA software tool that gives the user the ability to create and analyze fault trees and accident sequences using a microcomputer. This program provides functions that range from graphical fault tree construction to cut set generation and quantification. Also provided in the system is an integrated full-screen editor for use when interfacing with remote mainframe computer systems.

The INEL role in the IRRAS program is that of software developer and interface to the user community, including training and technology transfer. Version 1.0 of the IRRAS program was released in February of 1987 to prove the concept of performing this kind of analysis on microcomputers. This version contained many of the basic features needed for fault tree analysis and was received very well by the PRA community. Since the release of Version 1.0, many user comments and enhancements have been incorporated into the program, providing a much more powerful and user-friendly system. This version has been designated IRRAS 2.0.

IRRAS has all the capabilities and functions required to create, modify, reduce, and analyze fault tree models used in the analysis of complex systems and processes. IRRAS uses advanced graphic and analytical techniques to achieve the greatest possible realization of the potential of the microcomputer. When the needs of the user exceed this potential, IRRAS can call upon the power of the mainframe computer. Version 2.0 of IRRAS provides all of the same capabilities as Version 1.0 and adds a relational data base facility for managing the data, improved functionality, and improved algorithm performance.

ACKNOWLEDGMENTS

The IRRAS project would not have enjoyed the success it has without the efforts of some other key contributors. These individuals provided a major contribution to the development and success of the IRRAS system.

- Howard D. Stewart - for his support in designing the overall IRRAS concepts and for his software support in implementing these concepts.
- Dale M. Snider and Kurt L. Wagner - for their contribution to the graphical concepts and software development in both IRRAS 1.0 and the improved methods of IRRAS 2.0.

Acknowledgment is also given to the following personnel for their support on the IRRAS project.

- Doyle Batt - for his administrative expertise,
- Scott D. Cook, Donna J. Fink, Carol A. Mancuso, Scott D. Matthews, Matt K. McKay, and C. Wendell Richardson - for their programming support.

Appreciation is also expressed to all the various organizations and individuals who provided valuable comments and reviews of the preliminary versions of IRRAS, and special thanks go to the Beta Test site personnel.

CONTENTS

ABSTRACT	i
SUMMARY	iii
ACKNOWLEDGMENTS	iv
1. INTRODUCTION	1
2. GETTING TO KNOW IRRAS 2.0	4
2.1 IRRAS 2.0 Modules	4
2.2 IRRAS 2.0 Data Base Concepts	8
2.3 Other Things You Should Know	10
2.4 Getting Around Inside IRRAS 2.0	12
3. USING IRRAS 2.0	14
3.1 Getting Started	16
3.2 Define a Family	18
3.3 Build Fault Trees	22
3.3.1 Select a Family	22
3.3.2 Construct the Graphic Models	24
3.3.3 Cursor Control and Pop-up Menus	24
3.3.4 Fill Color	26
3.3.5 Fault Tree Modeling	27
3.3.5.1 Text.	29
3.3.5.2 Fault Tree Modeling - Part II.	32
3.3.5.3 Saving Our Work.	34
3.3.5.4 The Return to Fault Tree Modeling.	36
3.3.5.5 Modeling Symmetry	36
3.3.5.6 Saving a Previously Saved Drawing.	38
3.3.5.7 Saving a Drawing Under More Than One Name.	39
3.3.5.8 Modeling Initiating Events.	40
3.3.6 Assignment of Fault Tree Levels	41
3.4 Assigning Reliability Data	43
3.4.1 Failure Probability Calculation Types	46
3.4.2 Uncertainty Analysis Data	50

3.5	Analyzing Fault Trees	52
3.5.1	Generating Minimal Cut Sets and Quantifying the Fault Tree	54
3.5.2	Performing Fault Tree Uncertainty Analyses .	58
3.5.3	Displaying the Fault Tree Analysis Results .	61
3.5.4	Editing the Cut Sets	71
3.6	Identifying Accident Sequences	89
3.7	Sequence Analysis	94
3.7.1	Sequence Cut Set Generation and Quantification	94
3.7.2	Performing Sequence Uncertainty Analyses . .	99
3.7.3	Displaying the Sequence Analysis Results . .	101
3.8	Getting Reports on the Analysis	106
3.8.1	Basic Event Reports	107
3.8.1.1	Basic Events Summary Report	107
3.8.1.2	Basic Events Probabilities Report. .	110
3.8.1.3	Basic Events Uncertainty Values Report.	112
3.8.2	Fault Tree Reports	113
3.8.2.1	Fault Tree Logic Report.	114
3.8.2.2	Fault Tree Expanded Logic Report. . .	117
3.8.2.3	Fault Tree Modified Logic Report. . .	117
3.8.2.4	Fault Tree Importance Measures Report.	118
3.8.2.5	Fault Tree Cut Sets Report.	119
3.8.2.6	Fault Tree Summary Report.	122
3.8.2.7	Fault Tree Uncertainty Values Report.	123
3.8.3	Sequence Reports	125
3.8.3.1	Sequence Logic Report.	125
3.8.3.2	Sequence Summary Report.	126
3.8.3.3	Sequence Uncertainty Values Report. .	128
3.8.3.4	Sequence Importance Measures Report.	129
3.8.3.5	Sequence Cut Sets Report.	130
3.8.4	Graphics Printouts	133
3.8.4.1	Graphics on a Dot Matrix Printer. . .	133
3.8.4.2	Graphics on a Laser Printer.	133
3.8.4.3	Color Graphics on a Pen Plotter. . .	137

4. ADDITIONAL CAPABILITIES OF IRRAS 2.0	141
4.1 More on the Building of Fault Trees	141
4.1.1 Clearing Extracted Fault Trees	141
4.1.2 Extracting Fault Trees from the Data Base .	143
4.1.3 Build Graphic Trees	144
4.1.3.1 Basic Operation of Menus	145
4.1.3.2 Basic Fault Tree Symbols and Shapes .	147
4.1.3.3 Rules and Assumptions for Creating Fault Tree Logic	150
4.1.3.4 The Graphics Attribute Header	152
4.1.3.5 Main Graphics Menu	153
4.1.3.6 The BUILD Submenu	154
4.1.3.7 BUILD Attributes Submenu	157
4.1.3.8 The EDIT Submenu	159
4.1.3.9 The TEXT Submenu	160
4.1.3.10 The TEXT Attributes Submenu	165
4.1.3.11 The VIEWS Submenu	166
4.1.3.12 The SETUP Submenu	168
4.1.3.13 The FILE Submenu	171
4.2 Performing Sensitivity Analyses	174
4.2.1 Sensitivities Impacting the Reliability Data Base	174
4.2.1.1 Establishing a Base Case Data Base .	174
4.2.1.2 Creating a Sensitivity Analysis Data Base	178
4.2.1.3 Completing the Sensitivity Analysis .	182
4.2.2 Sensitivities Impacting the Logic Models . .	181
4.3 Adding Recovery Actions	183
4.4 Converting Studies Done on SETS to IRRAS	183
4.4.1 Converting SETS Input Files	183
4.4.2 Loading the Converted Files into the Data Base	188
4.4.3 Converting the Alphanumeric Logic File to Graphics	191
4.5 IRRAS 1.0/2.0 Interface	192
4.5.1 Converting from IRRAS 1.0 to IRRAS 2.0 . . .	192
4.5.1.1 Failure Rates	193
4.5.1.2 Fault Trees	194

4.5.1.3	Cut Sets	196
4.5.1.4	Graphics	198
4.5.2	Converting Data from IRRAS 2.0 to IRRAS	
1.0		199
4.5.2.1	Failure Rates	200
4.5.2.2	Fault Trees	201
4.5.2.3	Cut Sets	202
4.6	Define Constants	203
4.7	Communicating with a Mainframe Computer	210
4.8	More on the IRRAS/SETS Interface	211
4.9	Recovering the Data Base	216
4.10	Backing Up/Transferring the Data Base	216
4.11	Editing the Fault Trees	217
4.12	MAR-D Data Base Interface	240
4.12.1	Converting From IRRAS 2.0 Format to MAR-D	
	Format	241
4.12.1.1	Family	242
4.12.1.2	Basic Event	245
4.12.1.3	Fault Tree	255
4.12.1.4	Event Tree	264
4.12.1.5	Sequence	271
4.12.2	Converting from MAR-D format to IRRAS	
	format	279
4.12.2.1	Family	281
4.12.2.2	Basic Event	283
4.12.2.3	Fault Tree	293
4.12.2.4	Event Tree	300
4.12.2.5	Sequence	305
5.	FOR THE CURIOUS - A DISCUSSION OF ALGORITHMS	313
5.1	How Are Minimal Cut Sets Found?	313
5.1.1	Coalescing of Gates	313
5.1.2	Identifying Independent Subtrees	314
5.1.3	Treatment of Special Gates and Events	314
5.1.3.1	Treatment of N/M OR Gates	314
5.1.3.2	Treatment of Inhibit Gates	315

5.1.3.3	Treatment of Undeveloped Events and Transfers	315
5.1.3.4	Treatment of Transfer (TRANSIN) Gates	315
5.1.3.5	Treatment of HOUSE Events	316
5.1.3.6	Treatment of Complemented Events	316
5.1.3.7	Treatment of NOT Gates	317
5.1.4	Cut Set Generation	317
5.1.5	Supersetting	317
5.2	Minimal Cut Set Upper Bound	317
5.3	How Are the Sequence Cut Sets Determined?	318
5.4	Exploring Monte Carlo and Uncertainty	319
5.4.1	Basic Uncertainty Information	319
5.4.2	Supported Distributions	322
5.4.2.1	Beta Distribution	322
5.4.2.2	Gamma Distribution	322
5.4.2.3	Exponential Distribution	323
5.4.2.4	Normal Distribution	323
5.4.2.5	Lognormal Distribution	323
5.4.2.6	Uniform Distribution	323
5.5	What Are Importance Measures?	323
6.	REFERENCES	327
APPENDIX A--	HARDWARE REQUIREMENTS AND INSTALLATION PROCEDURE	A-1
APPENDIX B--	DATA BASE RELATIONS	B-1
APPENDIX C--	IRRAS 2.0 REPORTS	C-1
APPENDIX D--	IRRAS 2.0 MENUS AND FORMS	D-1
APPENDIX E--	CONVERSE COMMANDS	E-1
APPENDIX F--	DATA INTERCHANGE FORMATS	F-1

FIGURES

1.	IRRAS 2.0 modules and features.	4
2.	IRRAS 2.0 main menu.	5
3.	The MAIN and BUILD pop-up menus.	6
4.	IRRAS 2.0 data base structure.	7
5.	IRRAS Constants form for example.	11
6.	IRRAS Constants Graphics Information form.	12
7.	Key Functions Help screen.	13

8. Example problem piping and instrumentation diagram. . . .	14
9. Example problem event tree.	15
10. IRRAS Constants form for example.	17
11. IRRAS 2.0 main menu MODIFY option.	18
12. Edit Data Base Records Menu.	20
13. Family Editor menu.	20
14. DEMO FAMILY added to the data base.	21
15. Edit Data Base Records menu showing the DEMO FAMILY. . .	21
16. IRRAS-PC Fault Tree Graphics System menu (option S). . .	22
17. Select Family of Fault Trees menu.	23
18. The BUILD submenu.	26
19. Attributes submenu.	27
20. Text submenu.	29
21. ECS fault tree top event.	31
22. Partial ECS fault tree.	33
23. File submenu.	34
24. The Edit submenu.	37
25. Completed ECS fault tree.	38
26. CCS fault tree.	40
27. LOSP fault tree.	40
28. Edit Fault Tree menu.	42
29. Level and description added to the CCS fault tree. . . .	43
30. IRRAS-PC Fault Tree Analysis System menu.	44
31. Set Event Data menu.	45
32. Event Data menu.	46
33. Failure Data Calculation Types help menu.	47
34. Completed Event Data menu for basic event C-CV-A. . . .	51
35. Analyze Fault Trees menu.	54
36. Cut Set Generation Cutoff Values menu.	55
37. Batch Fault Tree Processing warning message.	56
38. CCS fault tree cut set generation summary.	57
39. Analyze Fault Trees menu after CCS cut set generation. .	58
40. Uncertainty Calculation Values menu.	59
41. Uncertainty results summary for the CCS fault tree. . . .	60
42. Display Fault Tree Data menu.	61
43. Fault Tree Cut Sets display.	62
44. Partition Cut Sets display.	63
45. Partition included events display.	64
46. View Events display.	65
47. Partitioned Cut Set display.	66
48. Partitioned Cut Set report.	67
49. Fault tree Cut Set Basic Event Descriptions display. . .	68
50. Fault tree Uncertainty Data display	69
51. Fault tree Basic Event Importance display.	70
52. Select a Cut menu.	71
53. Cut Set Editor menu and window.	72
54. Cut Set Editor cursor help menu.	74
55. Cut Set addition message.	75
56. Event addition message.	76
57. Addition to cut sets complete.	77
58. Modify cut set position.	78
59. Cut set number 1 moved to position 7.	79

60.	Modify event name message.	80
61.	Event name NEWEVENT changed.	81
62.	Insert message and prompt.	83
63.	The new event X added to three cut sets.	84
64.	Replace message and prompt.	85
65.	Result after the replace operation.	86
66.	Veto Delete menu.	88
67.	Options menu.	88
68.	Save Option menu.	88
69.	Edit Event Tree menu.	89
70.	LOSP Event Tree data.	90
71.	Edit Sequence menu.	91
72.	Sequence 1 added to the data base.	92
73.	Include Fault Trees menu for Sequence 1-LOSP-ECS.	92
74.	Logic for Sequence 1.	93
75.	IRRAS-PC Sequence Analysis System menu.	94
76.	Analyze Sequences menu.	95
77.	Sequence Cut Set Generation Cutoff Values menu.	96
78.	Cut set generation summary for Sequence 1-LOSP-ECS.	97
79.	Analyze Sequences menu after cut set generation.	98
80.	Sequence Uncertainty Calculation Values menu.	99
81.	Uncertainty results summary for Sequence 1-LOSP-ECS.	100
82.	Display Sequence Data menu.	101
83.	Sequence Cut Sets display.	102
84.	Sequence Cut Set Basic Event Descriptions display.	103
85.	Sequence Uncertainty Data display.	104
86.	Sequence Basic Event Importance display.	105
87.	The IRRAS Reports menu.	106
88.	IRRAS 2.0 Basic Event Reports menu.	107
89.	Basic Events Summary Report.	108
90.	IRRAS Reports Information Request form.	109
91.	IRRAS Reports Information Request form completed.	110
92.	Basic Events Probabilities Report.	111
93.	Basic Events Uncertainty Values Report.	112
94.	Fault Tree Reports menu.	114
95.	IRRAS Reports Information Request form for a Fault Tree Logic Report.	115
96.	Fault Tree Logic Report.	116
97.	Fault Tree Modified Logic Report for the CCS fault tree.	117
98.	Fault Tree Importance Measures Report.	118
99.	IRRAS Fault Tree Reports menu with Cut Sets Report Options.	120
100.	Fault Tree Cut Sets with Quantification Report.	121
101.	Fault Tree Summary Report.	122
102.	Fault Tree Uncertainty Values Report.	124
103.	Sequence Reports menu.	125
104.	Sequence Logic Report.	126
105.	Sequence Summary Report.	127
106.	Sequence Uncertainty Values Report.	128
107.	Sequence Importance Measures Report.	129

108.	Sequence Reports menu for the Sequence Cut Sets Report.	131
109.	Sequence Cut Sets with Quantification Report.	132
110.	IRRAS-PC Fault Tree Graphics System menu (option R).	134
111.	Rasterize Fault Trees menu.	135
112.	IRRAS Constants form for plotter communication port.	138
113.	Plot H.P. Fault Trees menu.	140
114.	IRRAS-PC Fault Tree Graphics System menu.	141
115.	Extract Fault Trees Menu.	142
116.	Display of extracted fault trees.	144
117.	Graphics module primary menu.	146
118.	EDIT submenu overlaying main graphics menu.	146
119.	The BUILD submenu.	147
120.	Fault tree basic shapes.	148
121.	Parent and child shape relationship.	150
122.	Arbitrary shape relationship.	150
123.	Relationships with intersecting lines.	151
124.	Attempting to jump lines.	151
125.	The graphics attribute list.	152
126.	The BUILD Attributes Submenu.	155
127.	The EDIT Submenu.	159
128.	The TEXT submenu.	161
129.	The TEXT Attributes submenu.	162
130.	The EDIT TABLE menu list.	164
131.	The VIEWS submenu.	166
132.	The SETUP submenu.	169
133.	The FILE submenu.	171
134.	Edit Data Base Records menu (option B).	175
135.	Edit Basic Event menu.	176
136.	Edit Event menu.	177
137.	Set Event Data menu with descriptions.	179
138.	Event Class Changes menu.	180
139.	Event Class Changes menu for check valve data change.	181
140.	SETS input file, CCS1.SET.	183
141.	IRRAS-PC Utilities menu (option I).	184
142.	IRRAS-PC IRRAS-to-SETS Interface menu (option X).	185
143.	IRRAS-to-SETS submenu.	186
144.	Convert From SETS to IRRAS menu for fault trees.	187
145.	SETS-to-IRRAS conversion output file, CCS1.TRE.	188
146.	IRRAS-PC Version 1.0/2.0 Interface menu.	188
147.	Version 1.0 to 2.0 conversion options.	189
148.	Convert Fault Tree Version 1.0 --> 2.0 menu (SETS).	190
149.	Convert Fault Trees menu.	191
150.	IRRAS-PC utilities menu (option V)	193
151.	Failure rate file format.	193
152.	Convert Event Data Version 1.0 --> 2.0 form.	194
153.	Fault tree logic file format.	195
154.	Convert Fault Tree Version 1.0 --> 2.0 form.	196
155.	Cut set file formats.	197
156.	Convert Cut Sets Version 1.0 --> 2.0 form.	198
157.	Convert Graphics Version 1.0 --> 2.0 form.	199
158.	Version 2.0 to 1.0 conversion options.	200

159.	Convert Event Data Version 2.0 --> 1.0 form.	201
160.	Convert Fault Tree Version 2.0 --> 1.0 form.	202
161.	Convert Cut Sets Version 2.0 --> 1.0 form.	203
162.	IRRAS-PC Utilities menu (option D).	204
163.	IRRAS Constants form (option D).	204
164.	IRRAS Constants Graphics Information form (option D). .	208
165.	IRRAS-PC IRRAS-to-SETS Interface menu (option I). . . .	212
166.	IRRAS-to-SETS submenu.	213
167.	IRRAS-to-SETS Convert Event Data submenu.	213
168.	IRRAS-to-SETS Fault Tree menu.	214
169.	MFRAME.TRE logic file.	214
170.	SETS input file MFRAME.SET.	215
171.	Recover Data Base warning.	216
172.	Select a Fault Tree menu.	217
173.	Fault Tree Editor menu and window.	218
174.	Fault Tree Editor cursor help menu.	219
175.	Beginning a Gate Addition.	220
176.	Completion of a Gate Addition.	221
177.	Addition of input to a gate.	222
178.	Gate or Event inquiry.	223
179.	Entering an unknown Gate.	224
180.	Entering unknown events.	225
181.	Modification of a gate, type, or input.	226
182.	Locate a gate or input	227
183.	Search inputs for a string.	228
184.	Beginning a Find and Insert.	229
185.	Completion of a Find and Insert.	230
186.	Beginning of a Find and Replace.	231
187.	Completion of a Find and Replace.	232
188.	Help Message for Find and Insert/Replace.	232
189.	Completion of a Delete Gate.	233
190.	Completion of a Delete Input.	234
191.	The Fault Tree Editor Options Menu.	235
192.	The Veto Message.	236
193.	Help Message for Find and Insert/Replace.	236
194.	The Help Menus Turned Off.	237
195.	Help Message for Hide Menu.	237
196.	Check and save data inquiry.	238
197.	Legal gate types.	239
198.	UTILITY Options Menu.	241
199.	IRRAS-PC IRRAS / MAR-D Interface menu.	241
200.	IRRAS to MAR-D Conversion Options menu.	242
201.	IRRAS to MAR-D Family Conversion Options menu.	243
202.	IRRAS to MAR-D Family Name and Description Conversion menu.	244
203.	MAR-D Family Name and Description File Format.	244
204.	IRRAS to MAR-D Family Text Conversion menu.	245
205.	MAR-D Family Text File Format.	245
206.	IRRAS to MAR-D Basic Event Conversion Options.	246
207.	IRRAS to MAR-D Basic Event Name and Description Conversion Menu.	247
208.	MAR-D Event Name and Description File Format.	247

209.	IRRAS to MAR-D Basic Event Probability Conversion Menu.	247
210.	MAR-D Probability File Format.	248
211.	IRRAS to MAR-D Basic Event Attribute Conversion Menu.	249
212.	MAR-D Attribute File Format.	249
213.	IRRAS to MAR-D Basic Event Continuation Menu.	250
214.	IRRAS to MAR-D Basic Event Failure Conversion menu.	250
215.	MAR-D failure file format.	251
216.	IRRAS to MAR-D Basic Event Component Conversion menu.	251
217.	MAR-D Component File Format.	252
218.	IRRAS to MAR-D Basic Event System Conversion menu.	252
219.	MAR-D System File Format.	253
220.	IRRAS to MAR-D basic event location conversion menu.	253
221.	MAR-D location file format.	254
222.	IRRAS to MAR-D basic event attribute class conversion menu.	254
223.	MAR-D attribute class file format.	255
224.	IRRAS to MAR-D fault tree conversion options menu.	255
225.	IRRAS to MAR-D fault tree name and description conversion menu.	256
226.	MAR-D fault tree description file format.	256
227.	IRRAS to MAR-D fault tree graphics conversion menu.	257
228.	IRRAS to MAR-D fault tree logic conversion menu.	258
229.	MAR-D fault tree logic file format.	259
230.	IRRAS to MAR-D fault tree cutset conversion menu.	260
231.	MAR-D fault tree cutsets file format.	261
232.	IRRAS to MAR-D fault tree attribute conversion menu.	262
233.	IRRAS to MAR-D fault tree attribute file format.	263
234.	IRRAS to MAR-D fault tree test conversion menu.	264
235.	MAR-D fault tree test file format.	264
236.	IRRAS to MAR-D event tree conversion options menu.	265
237.	IRRAS to MAR-D event tree name and description conversion menu.	266
238.	MAR-D event tree description file format.	266
239.	IRRAS to MAR-D event tree graphics conversion menu.	267
240.	IRRAS to MAR-D event tree logic conversion menu.	268
241.	IRRAS to MAR-D event tree attribute conversion menu.	269
242.	MAR-D event tree attribute file format.	269
243.	IRRAS to MAR-D event tree text conversion menu.	270
244.	MAR-D fault tree text file format.	270
245.	IRRAS to MAR-D sequence conversion options menu.	271
246.	IRRAS to MAR-D sequence name and description conversion menu.	272
247.	MAR-D sequence description file format.	272
248.	IRRAS to MAR-D sequence logic conversion menu.	273
249.	IRRAS to MAR-D sequence cutsets conversion menu.	274
250.	MAR-D sequence cutsets file format.	275
251.	IRRAS to MAR-D sequence attribute conversion menu.	276
252.	MAR-D sequence attribute file format.	277
253.	IRRAS to MAR-D sequence text conversion menu.	278
254.	MAR-D sequence attribute file format.	278
255.	UTILITY options menu.	279

256.	IRRAS-PC IRRAS / MAR-D Interface menu.	280
257.	MAR-D to IRRAS conversion options menu.	280
258.	MAR-D to IRRAS family conversion options menu.	281
259.	MAR-D to IRRAS family name and description conversion menu.	282
260.	MAR-D to IRRAS family text conversion menu.	283
261.	MAR-D to IRRAS basic event conversion options menu. . .	284
262.	MAR-D to IRRAS basic event description conversion menu.	285
263.	MAR-D to IRRAS basic event probability conversion menu.	286
264.	MAR-D to IRRAS basic event attribute conversion menu. .	287
265.	MAR-D to IRRAS basic event continuation menu.	288
266.	MAR-D to IRRAS basic event failure conversion menu. . .	289
267.	MAR-D to IRRAS basic event component conversion menu. .	290
268.	MAR-D to IRRAS basic event system conversion menu. . .	291
269.	MAR-D to IRRAS basic event location conversion menu. .	292
270.	MAR-D to IRRAS basic event attribute class conversion menu.	293
271.	MAR-D to IRRAS fault tree conversion options menu. . .	294
272.	MAR-D to IRRAS fault tree name and description menu. .	295
273.	MAR-D to IRRAS fault tree graphics conversion menu. . .	296
274.	MAR-D to IRRAS fault tree logic conversion menu. . . .	297
275.	MAR-D to IRRAS fault tree cutsets conversion menu. . .	298
276.	MAR-D to IRRAS fault tree attribute conversion menu. .	299
277.	MAR-D to IRRAS fault tree text conversion menu.	300
278.	MAR-D to IRRAS event tree conversion options menu. . .	301
279.	MAR-D to IRRAS event tree name and description conversion menu.	302
280.	MAR-D to IRRAS event tree graphics conversion menu. . .	303
281.	MAR-D to IRRAS event tree logic conversion menu. . . .	304
282.	MAR-D to IRRAS event tree attribute conversion menu. .	305
283.	MAR-D to IRRAS event tree text conversion menu.	306
284.	MAR-D to IRRAS sequence conversion options menu. . . .	307
285.	MAR-D to IRRAS sequence name and description conversion menu.	308
286.	MAR-D to IRRAS sequence logic conversion menu.	309
287.	MAR-D to IRRAS sequence cutsets conversion menu. . . .	310
288.	MAR-D to IRRAS sequence attribute conversion menu. . .	311
289.	MAR-D to IRRAS sequence text conversion menu.	312
290.	Coalescing of OR gates.	313
291.	Conversion of a 2-out-of-3 gate.	314
292.	Example use of an INHIBIT gate.	315
D-1.	Hierarchy of IRRAS 2.0 menus, forms, and options . . .	D-4

TABLES

1.	Example problem data base	15
2.	IRRAS Constants Form Field Descriptions	206
3.	IRRAS Constants Graphics Information Form Field Descriptions	209
4.	Distributions and Parameters	322

E-1. CONVERSE System Parameters E-3
E-2. CONVERSE Key Descriptions E-5

INTEGRATED RELIABILITY AND RISK ANALYSIS SYSTEM
(IRRAS) VERSION 2.0
USER'S GUIDE

1. INTRODUCTION

The Integrated Reliability and Risk Analysis System (IRRAS) software development project was started as a result of a recognized need for microcomputer-based software to aid the probabilistic risk assessment (PRA) analyst. The initial scope of the project was to provide a software package which could demonstrate the feasibility of using the microcomputer as a workstation for performing PRA analyses. This package did not necessarily need to perform all of the functions required; however, it did need to provide certain essential functions such as fault tree construction, failure data input, cut set generation, and cut set quantification. The result of this software development project was IRRAS 1.0. This version of the software was released in February of 1987 and contained only the essential concepts mentioned above.

IRRAS 1.0 was an immediate success and clearly demonstrated not only the tremendous need but also the feasibility of performing this work on a microcomputer. As a result of this success, IRRAS 2.0 development was begun. This package was designed to be a comprehensive PRA analysis package and include all the functions necessary for a PRA analyst to perform his work. The areas which were not treated in version 1.0 were addressed, and a complete integrated package was developed. Since IRRAS version 2.0 was a complete rewrite from version 1.0, a thorough test plan was necessary. The major features of IRRAS 2.0 along with an Alpha test were completed in early March of 1988. Following the Alpha test, approximately 15 sites were selected from among the sites currently using IRRAS 1.0. and were sent a Beta test version of IRRAS 2.0. In May of 1988, we completed the Beta test and began work on fixing any bugs found and including those desired new features which could reasonably be incorporated into version 2.0. IRRAS 2.0 is now ready for distribution.

IRRAS version 2.0 is an integrated PRA software tool that gives the user the ability to create and analyze fault trees and event trees using a personal computer (PC). This program provides functions for fault tree and event tree construction and analysis. The fault tree functions range from graphical fault tree construction to fault tree cut set generation and quantification. The event tree functions include the linking of fault trees, defining accident sequences, generating accident sequence cut sets, and quantifying them.

The objective of the IRRAS development is to provide the U.S. Nuclear Regulatory Commission (NRC) with a PC-based tool that can be used by NRC staff to evaluate incidents in a timely manner.

IRRAS 1.0 was released in February 1987. Since that time, it has been distributed to over 250 users in the United States. These users include NRC staff, national laboratories, contractors, vendors, utilities, architectural engineering firms, consultants, universities, other government agencies, and their contractors. IRRAS is being used as an analysis tool in the resolution of generic issues, inspection activities, and other NRC programs.

At the center of the PRA is the fault tree model. This model, along with component reliability data, provides the basis for risk analysis in most PRA methods. Prior to the development of IRRAS, an analyst would generate models using an alphanumeric text editor. The information would be input in a format compatible with the analysis software, such as SETS, used to process the information. When changes to the model were needed, the analyst would go back to the text file and modify the card images to reflect the changes in the logic model. He would then resubmit them to generate the new cut sets. This process is prone to errors and difficult for the analyst to visualize. IRRAS helps eliminate this problem.

IRRAS automates the model creation, manipulation, modification, and quantification processes. Designed for the IBM-PC^a and compatibles, IRRAS is readily accessible and portable. Taking advantage of analysis algorithms developed at the Lawrence Livermore National Laboratory, IRRAS is quite fast and powerful. For problems that are too large to be solved on the PC, IRRAS communicates with a mainframe computer.

IRRAS simplifies the analysis process and automates the construction of input to the analysis software. The analyst can graphically construct and modify fault trees. IRRAS gives the users better visualization of the fault tree and simplifies the construction and maintenance. The program supports all of the basic constructs involved in fault tree construction, including NOT gates. Once the fault tree is constructed, the program automatically generates the alphanumeric input for the analysis software. The component reliability information is then easily input into the IRRAS data base using specially designed menus and screens.

After constructing the system fault tree models, the analyst processes these models with an integrated fault tree analysis package. This package includes the capability to read a fault tree and failure rate data associated with the basic events. The program then generates the fault tree minimal cut sets of the fault tree and quantifies the fault tree top event probability using the

a. Mention of specific products and/or manufacturers in this document implies neither endorsement or preference nor disapproval by the U.S. Government, any of its agencies, or EG&G Idaho, Inc., of the use of a specific product for any purpose.

minimal cut set upper bound. Importance measures for both cut sets and basic events are calculated. The results are documented in various reports generated by the program. The user may select truncation of cut sets by size and/or probability and specify the gate where reduction is to begin. The user may perform some cut set level analysis by using a cut set editor to modify the cut sets, saving the new cut sets, and recalculating the minimal cut set upper bound and the new importance measures.

The analyst defines accident sequences in terms of the systems. IRRAS 2.0 has the ability to link fault trees according to the accident sequence logic to create core damage sequence cut sets. These accident sequence cut sets are then quantified to determine the accident sequence frequency. Importance measures are also calculated.

Many features of mainframe codes have been incorporated into IRRAS 2.0. Improved fault tree reduction techniques, such as identification of independent subtrees and coalescing gates, help to speed up the analysis and make it possible to process larger logic models. Error-checking routines have been added to ensure that the input can be processed when it is time to analyze the fault trees.

In IRRAS 2.0, the graphical fault tree logic can be directly generated from alphanumeric input. This allows the user to read mainframe code input files, such as SETS, and generate the fault tree graphics. The logic models are then easily modified for re-analysis.

IRRAS 2.0 includes fault tree, event tree and cut set editors to improve the analysis capabilities without requiring complete regeneration and reduction of the fault trees. Basic event or initiating event frequencies are easily changed. Cut sets are easily modified with the cut set editor to add recovery actions, or cut sets may be deleted if desired. These changes can be saved in the data base and quantified as desired.

The sections that follow provide the user with the detailed information needed to become familiar with IRRAS 2.0. Sections 2 and 3 present a step-by-step approach to using the basic capabilities of IRRAS, and Section 4 presents more details of how IRRAS works. Code algorithms are discussed in Section 5.

Appendix A provides the specifics of hardware requirements and installation; Appendices B through E provide a more detailed treatment of data base relations and other supporting documentation.

2. GETTING TO KNOW IRRAS 2.0

The purpose of this section is to give you, the user, an overview of what IRRAS 2.0 can do. We begin with brief discussions of the IRRAS 2.0 modules. (Each module roughly corresponds to a major function or feature.) Next, we explain a little about the IRRAS 2.0 data base so you can get a feeling for what happens to the information you put in IRRAS 2.0. And finally, we give some pointers on how to best get around inside the program with some special keys. We recommend that you follow along on your PC as you go through the examples in this manual.

2.1 IRRAS 2.0 Modules

We have structured IRRAS 2.0 so that the various functions are contained in individual modules or program units. Each module is activated by the use of a command from a menu on the screen. The main modules are BUILD Fault Trees, FAULT Tree Analysis, CREATE Event Trees, SEQUENCE Analysis, MODIFY Data Base, REPORT on Data Base, and UTILITY Options. Figure 1 summarizes the main modules and their functions. The main menu is shown in Figure 2.

Figure 1. IRRAS 2.0 modules and features.

Figure 2. IRRAS 2.0 main menu.

The BUILD Fault Tree module is used to draw fault tree models and print them out. Fault trees can be drawn using the keyboard, a graphics tablet, or a mouse. If you use the keyboard, you'll soon see the advantages of getting a tablet or mouse. We have used pop-up menu technology in this module to make tree drawing as simple as possible. Figure 3 shows an example of the MAIN and BUILD pop-up menus.

Figure 3. The MAIN and BUILD pop-up menus.

The FAULT Tree Analysis module is used to quantify the fault tree. Here you determine the minimal cut sets, calculate the probability of failure of the top event of the fault tree, calculate the importance measures of the basic events and the cut sets, and perform uncertainty analysis on the top event failure probability. Sensitivity analysis can be done by changing the failure probability of basic events and requantifying.

The CREATE Event Tree option will be implemented in version 3.0 of IRRAS. This option will provide the user with a tool to graphically construct event trees and link them similar to the way the fault trees are currently constructed in IRRAS 2.0. This should greatly simplify and automate the event tree construction and analysis process.

In the meantime, we have provided a powerful sequence analyzer in the SEQUENCE Analysis module. This module generates the cut sets for an accident sequence. This is done by combining the cut sets for the fault trees that make up the sequence. Comparisons are made between the cut set lists for successful and failed fault trees to eliminate impossible failure combinations (similar to the delete term option in SETS). Once the sequence cut sets have been generated, you can quantify the sequence, calculate importance measures, and perform uncertainty analysis. The results

can be displayed on the screen, sent to a printer, or written to a file for later use.

The MODIFY Data Base module provides access to the IRRAS 2.0 relational data base that is created, in part, from use of the other modules. Figure 4 shows the structure of this data base. All information is grouped into families. A family can be a facility, a group of systems, or any other logical group for which fault trees or sequences are desired. You define the families and assign descriptions in this module. Fault trees are assigned to families in the BUILD Fault Trees module. Sequences are created and assigned descriptions in the MODIFY Data Base module. Basic events are automatically extracted from the fault trees assigned to the family, but you must assign reliability data and descriptions in the MODIFY Data Base module. All additions, deletions, and modifications to the data base are made here. Section 2.2 and Appendix B provide more detailed information on the data base.

Figure 4. IRRAS 2.0 data base structure.

The REPORT on Data Base module lets you see what is in the IRRAS 2.0 data base. Reports are available on families, basic events, fault trees, and sequences. All the information shown on Figure 4 is contained in these reports. Appendix C shows the format for all the available reports.

The last module, UTILITY Options, performs program and computer-related tasks such as setting new machine constants after a hardware configuration change or interfacing with SETS or the PRA data base.

2.2 IRRAS 2.0 Data Base Concepts

Before you can use IRRAS 2.0 effectively, you need a good understanding of the IRRAS 2.0 data base structure and concepts. The data base provides an effective way to manage the inputs and outputs of the various modules.

The IRRAS 2.0 data base is divided into families. A family is the first thing that must be created when using IRRAS 2.0. As delivered, IRRAS 2.0 provides an initial family called DEMO FAMILY. (Once you are comfortable with IRRAS 2.0, you may want to clear this family from the data base.) A family is any logical grouping of fault trees and sequences with their associated basic events, cut sets, reliability data, and descriptions. Access to any portion of the data base is obtained through the appropriate family.

A fault tree is automatically assigned to the current family selected when the BUILD Fault Trees module is exited. The basic events used by the fault tree are checked against the events already assigned to the family. Only new basic events are added to the family list. Thus, for a given family there is only one list that contains all the basic events for all the fault trees in the family. This assures that the reliability data used for a basic event in one tree are identical to those used for the same event in another tree in the family.

The family relation maintains data common to all fault trees in the family, including

- o family name and description,
- o basic event names and attributes,
- o event-to-fault tree association,
- o event-to-sequence association,
- o fault tree change status information, and
- o sequence change status information.

The fault tree relation maintains data for the following fault tree attributes:

- o fault tree name and description,
- o fault tree logic,
- o graphic representation,
- o minimal cut sets,
- o top event probability,
- o importance measures, and
- o uncertainty analysis results.

Similarly, the relation defined for an accident sequence includes:

- o name and description,
- o logic (combination of successful and failed systems),
- o minimal cut sets,
- o sequence probability,
- o importance measures, and
- o uncertainty analysis results.

The event tree relation includes:

- o event tree name and description,
- o graphic representation (not yet implemented), and
- o initiating event (not yet implemented).

A relation is also maintained to track event changes relating to event class and event probability.

Unrelated to trees and /or families is the user/system data.

The fore-mentioned relations are permanent relations used to maintain the IRRAS 2.0 data base. For each relation, three files exist:

- o *.DAT,
- o *.IDX, and
- o *.BLK.

The *.DAT file contains the actual data and information about which data constitutes a "key". The *.IDX file contains indices into the data for keyed fields. Finally, the *.BLK file contains variable length data (such as cut sets) that are accessible through pointers contained in the *.DAT file.

Appendix B contains a detailed report of the contents of each relation in the data base, including temporary relations (file number = 0).

2.3 Other Things You Should Know

The installation procedure for loading IRRAS 2.0 into your computer is found in Appendix A. The loading process creates three new directories if they do not already exist. These are \IRRAS2, \IRRAS2\DEMO, and \HALO. The IRRAS2 directory contains the program, the \IRRAS2\DEMO directory contains sample data, and the HALO directory contains the graphics package. These directories take approximately 2500K of memory on the hard disk when first loaded. Of course, storage requirements go up as you use the program and generate more data.

IRRAS 2.0 uses approximately 600K of internal memory when executed, therefore, resident programs such as Sidekick¹ must be unloaded before using IRRAS 2.0 on a 640K machine. Also, having the computer on a network can take up enough internal memory to cause problems.

The following procedure assumes your system is running DOS 3.0 or newer. If your DOS version is older than 3.0, see Appendix A.

To start IRRAS 2.0, simply change directories to IRRAS2 (CD\IRRAS2), type IRRAS2, and hit <Enter>. (The <> designates depress the indicated key.) This will give you IRRAS 2.0 with the keyboard as the graphics input device. To use the mouse, type IRRAS2 M <Enter>. If the mouse is connected to communications port 2, then type IRRAS2 M 2 <Enter>. To use a graphics tablet, type in the commands as you would for a mouse and substitute T for M. See Appendix A for more on entering the program. The .BAT files provided with the IRRAS 2.0 program may be placed in a BATCH directory if desired.

On the very first entry into IRRAS 2.0 after installation, the IRRAS Constants form will appear. This form is shown in Figure 5. With this form, you tell the program the configuration of your hardware and the default values desired for various functions within the program. This form, once completed, will not appear on subsequent entries into IRRAS. If you desire to change a default or your hardware configuration has changed, this form can be called up from the UTILITY Options menu (see Section 4.6.1).

I R R A S C o n s t a n t s			
User Information			
User Name	EG&G Idaho, Inc. (INEL)		
Monitor type (0-EGA, 1-CGA)			0
Locator type (1-Keyboard/Tablet/Logitech, 2-Microsoft)			1
Printer has complete IBM character font? (Y/N)			N
Plotter communications port (0-None, 1-Com1, 2-Com2, 3-Com3)			0
Use primary basic event Names ? (Y=primary, N=alternate)			Y
Use ratios in Importance Measures ? (Y=ratios, N=intervals)			Y
Uncertainty			
Random number seed for uncertainty calculations			3571
Sample size			1000
Cut Set Generation			
Cutoff by size? (Y/N)	Y	Size cutoff	6
Cutoff by probability? (Y/N)	Y	Probability cutoff	1.000E-015
Mission time (hours)			2.400E+001
File Locations			
Halo drive and directory path	\HALO\		

Change any of the constant values shown and press the <Enter> key.

Figure 5. IRRAS Constants form for example.

When this form is filled out properly, press <Enter> to place the information in the data base. Another defaults/constants screen will appear. This screen, shown in Figure 6, establishes the defaults for the fault tree graphics.

I R R A S C o n s t a n t s			
Graphics Information			
Names			
Event Name	EVENT	Name Height	.50
Gate Name	GATE	Line Type	1
Text			
Height	1.00	Justification	C Box Y
N/M Height	1.50	Space Factor	1.40
Toggles		Colors	
Grid	N	Colors ==>	
Blank	Y	Fill Color	X
Show Name	Y	Name Color	X
Show Text	Y	Line Color	X
MultiPick	Y	Text Color	X
Change any of the constant values shown and press the <Enter> key.			

Figure 6. IRRAS Constants Graphics Information form.

2.4 Getting Around Inside IRRAS 2.0

Option selection from any menu with a vertical list of options can be done in several ways. The up and down arrow keys can be used to move a highlight through the options. When the highlight is on the desired option, press <Enter> to select the option. The <Space bar> will also move the highlight through the options. Another method of selecting an option is to type in the first letter of the option and pressing <Enter>. On menus that have horizontal lists of options, the first letter of the option must be typed in the option field followed with <Enter>.

Certain portions of the program have help screens as aids to the user. These help screens are accessed by pressing the <Esc> key.

Pressing <Alt> H causes the Key Functions help screen to appear. This screen is shown in Figure 7. This screen identifies the functions of some special keys for use on fill-in-the-blank type forms used in IRRAS 2.0.

Key Functions Help	
exit-----Enter	restore field-Ctl/R
home-----Home	clear form----Ctl/Home
get help-ESC	clear field---Ctl/End
end help-F10	copy screen---Ctl/K
key help-Alt/H	B&W / color---Ctl/B
insert---Ins	delete-----Del

Figure 7. Key Functions Help screen.

The carriage return <CR> or <Enter> key sends a completed form to the program for execution. The arrow keys and <Tab> key move the cursor around the fields on a form. The <Home> key moves the cursor to the home field on the form. The <Ins> key inserts characters in the text in a field. The <F10> key drops a help screen. <Ctrl> R restores a field to its original setting. <Ctrl> <Home> clears all entries in all fields on the form. <Ctrl> <End> clears the selected field. <Ctrl> B switches the display between black and white and color. <Ctrl> K copies the screen into a file on the disk named SCREEN.CPY. Multiple uses of <Ctrl> K will cause each screen to be appended to the bottom of the SCREEN.CPY file.

3. USING IRRAS 2.0

Now that you are somewhat familiar with what IRRAS 2.0 can do, we will show you how IRRAS 2.0 can be used to perform some basic analyses. More advanced capabilities will be discussed later, but for now we'll stick to the basics.

This discussion will be centered around an example problem that will go through all the steps needed to perform fault tree and sequence analyses.

Our task is simply stated: Model the systems shown in Figure 8 and determine the probability of core melt due to the failure of these systems in a loss of offsite power situation.

Figure 8. Example problem piping and instrumentation diagram.

In our example, the event tree of Figure 9 has already been developed. This defines the accident sequences we need to analyze. Component reliability data and initiating event data have already been determined and are provided in Table 1.

Figure 9. Example problem event tree.

COMPONENT TYPE	FAILURE PROBABILITY MEAN VALUE	LOGNORMAL DISTRIBUTION ERROR FACTOR
Check Valve	1.0E-04	3.0
Pump discharge isolation MOV	5.0E-03	5.0
Tank isolation MOV	1.0E-03	5.0
Motor-driven pump	3.0E-03	5.0
Storage tank	1.0E-07	3.0
Diesel generator	2.0E-02	10.0
INITIATING EVENT	FREQUENCY	
LOSP	2.3/yr	10.0

Table 1. Example problem data base.

NOTE: This example problem is for demonstration of IRRAS 2.0 capabilities only and does not

present all the engineering, such as transient analysis and data collection, needed for a PRA analysis.

3.1 Getting Started

Directory \IRRAS2\DEMO contains all of the data base files used in Section 3.0. If you wish to re-enter all of the information as you follow along with the manual, change directory to \IRRAS2 and type

```
IRRAS2 M <Enter>
```

If you wish to follow along with the existing data, change directory to \IRRAS2 and type

```
copy demo\*. * <Enter>
```

NOTE: This will destroy any existing IRRAS 2.0 data base files you may have created.

Then execute IRRAS 2.0 by typing

```
IRRAS2 M <Enter>
```

or if you have DOS 3.0 or greater, change directory to \IRRAS2\DEMO and type

```
IRRASA M <Enter>
```

The M indicates that we have a mouse as our graphics input device. Appendix A explains the commands for other hardware configurations and other directories.

NOTE: The data base supplied in the DEMO directory has been completely run; therefore, some menu flags may not exactly match those described in the example.

Because this was our first time using IRRAS 2.0, the IRRAS Constants form (See Figure 10) was automatically brought up. This form requires that the following fields be completed:

FIELD	DESCRIPTION
Monitor type	0 - Enhanced graphics monitor and card 1 - Standard color graphics monitor and board
Locator type	1 - Keyboard/Tablet/Logitech 2 - Microsoft
Printer has complete IBM character font?	Y - Yes N - No
Plotter communications port	0 - No plotter connected to port 1 - Plotter connected to Com1 port 2 - Plotter connected to Com2 port 3 - Plotter connected to Com3 port

All other fields are optional or may be completed at the discretion of the user. If a plotter or printer is not to be used, then the fields pertaining to these components are of no significance. More information on the use of the IRRAS Constants form is provided in Section 4.6. Once you have supplied the needed information and press <Enter>, the IRRAS 2.0 main menu appears. All future entries into the IRRAS 2.0 program will begin with the main menu. Figure 10 shows the constants set for our example problem.

I R R A S C o n s t a n t s			
-----User Information-----			
User Name	EG&G Idaho, Inc. (INEL)		
Monitor type (0-EGA, 1-CGA)			0
Locator type (1-Keyboard/Tablet/Logitech, 2-Microsoft)			1
Printer has complete IBM character font? (Y/N)			N
Plotter communications port (0-None, 1-Com1, 2-Com2, 3-Com3)			0
Use primary basic event Names ? (Y=primary, N=alternate)			Y
Use ratios in Importance Measures ? (Y=ratios, N=intervals)			Y
-----Uncertainty-----			
Random number seed for uncertainty calculations			0
Sample size			1000
-----Cut Set Generation-----			
Cutoff by size? (Y/N)	Y	Size cutoff	6
Cutoff by probability? (Y/N)	Y	Probability cutoff	1.000E-015
Mission time (hours)			2.400E+001
-----File Locations-----			
Halo drive and directory path	\HALO\		

Figure 10. IRRAS Constants form for example.

3.2 Define a Family

The first step in our new analysis was to create a family in which all the data would go. We did this by selecting the MODIFY Data Base option from the IRRAS 2.0 main menu, as shown in Figure 11. Option selection from this menu, and all other menus, can be done in any of several ways. The up and down arrow keys can be used to move a highlight through the options. When the highlight is on the desired option, press <Enter> to select the option. The <Space Bar> will also move the highlight through the options. Another method of selecting an option is to type in the first letter of the option and pressing <Enter>.

Figure 11. IRRAS 2.0 main menu MODIFY option.

Selecting MODIFY Data Base from the main menu brought up the Edit Data Base Records menu shown in Figure 12. Since we wanted to work with families, we selected the Family option by typing F <Enter>. This brought up the Family Editor menu, as shown in Figure 13. The Add option was chosen by typing A. The highlight moved to the Family Name field, where we typed in the desired family name. (In this case, we chose Demo Family; however, the names you choose for families should be descriptive of the contents, and the word family need not appear in the name.) We then used the <Down Arrow> key to move the highlight down to the Family Description field and typed Demonstration sample family as our description. The screen then looked like the menu shown in Figure 14. Hitting the <Enter> key at this time caused a family called Demo Family, along with its description, to be added to the data base. This was indicated by the computer writing to the hard disk and the message "Record Added" appearing at the bottom of the screen.

Having finished with this menu, we typed E <Enter> to exit back to the Edit Data Base Records menu. This menu then showed the newly created family (See Figure 15). No more data base editing was desired, so we typed E <Enter> to exit back to the IRRAS 2.0 main menu.

NOTE: A family must be added to the data base and selected before any other functions can be performed. This should be the first operation performed on an empty data base.

E d i t D a t a B a s e R e c o r d s

Option |E| Exit / Family / Basic Events
 1 - Fault Trees / 2 - Event Trees & Sequences

Note: Select type of data to be edited, move cursor to desired family, and press <Enter>.

Use <PgUp> or <PgDn> to display more Families.

Figure 12. Edit Data Base Records Menu.

F A M I L Y E D I T O R

Option |E| Exit / Add / Modify / Delete

Family Number ----

Family Name NO FAMILY

Sequence Sum -----E----

Family Description

Note: The Add options adds a new family record. The Modify and Delete options edit the displayed family record.

Figure 13. Family Editor menu.

F A M I L Y E D I T O R

Option |A| Exit / Add / Modify / Delete

Family Number ----

Family Name DEMO FAMILY

Sequence Sum -----E----

Family Description

Demonstration sample family

Note: The Add options adds a new family record. The Modify and Delete options edit the displayed family record.

Figure 14. DEMO FAMILY added to the data base.

E d i t D a t a B a s e R e c o r d s

Option |E| Exit / Family / Basic Events
 1 - Fault Trees / 2 - Event Trees & Sequences

DEMO FAMILY Demonstration sample family

Note: Select type of data to be edited, move cursor to desired family, and press <Enter>.

Use <PgUp> or <PgDn> to display more Families.

Figure 15. Edit Data Base Records menu showing the DEMO FAMILY.

3.3 Build Fault Trees

3.3.1 Select a Family

With a family created, we were ready to begin building fault tree models of our system. From the IRRAS 2.0 main menu, we selected the BUILD Fault Trees option. The IRRAS-PC Fault Tree Graphics System menu came up and indicated that no family was currently selected (see Figure 16). If a family had been selected earlier in this or some other module, that family would remain as the currently selected family until it was changed.

Figure 16. IRRAS-PC Fault Tree Graphics System menu (option S).

To select the Demo Family, we typed S <Enter> for the Select Family option and the Select Family of Fault Trees menu was displayed. The highlight was on Demo Family and option S for Select Family was already chosen, so we hit <Enter> to change the currently selected family to Demo Family. To confirm that a family has been selected, the computer displays an informational message at the bottom of the screen stating Family Selected. Figure 17 shows what the menu looked like at this point. Pressing <Enter> also changed the menu option from Select Family to Exit, therefore, we only had to hit <Enter> again to get back to the Fault Tree Graphics System menu.

Select Family of Fault Trees

Currently Selected Family —> DEMO FAMILY

Option |E| Exit / Select Family

DEMO FAMILYDemonstration sample family

NOTE: To select a Family, enter option <S>, move the cursor to the desired Family, press <Enter>. Use <PgUp> or <PgDn> to display more families.

Figure 17. Select Family of Fault Trees menu.

3.3.2 Construct the Graphic Models

We selected the Build Graphics Trees option on the Fault Tree Graphics System menu (Figure 16). The screen went blank and then a single blue line with information appeared at the top and a green cross appeared in the middle of the screen. This was our drawing surface and drafting equipment. The blue information line gives us the status of various drawing options and is discussed in Section 4.1.3.4. The cross is called a cursor and is controlled by the graphics input device, in our case, a mouse. The cursor is used to position pop-up menus, draw lines, place drawing symbols, and select menu options.

NOTE: If the screen stays blank and the keys do not have any affect, the type of monitor was probably incorrectly specified. Reboot the computer and refer to Section 3.1.

3.3.3 Cursor Control and Pop-up Menus

Using the mouse, we moved the cursor off to one side to give us plenty of drawing space and pressed the left button to bring up the main Graphics menu. The menu pops up with its upper-left-hand corner on the cursor location.

NOTE: Cursor control with a drawing tablet is identical to that with a mouse. Cursor control with the keyboard uses the arrow keys for position control, and the <z> key to perform the functions of the right button of a mouse, and the left shift corresponds as the left button on the mouse. In order to use the keyboard after utilizing the arrow keys, the ESC key must be used as a toggle switch in order to restore normal keyboard functions (i.e. for text entry). To restore control back to the arrow keys after utilizing the keyboard, the ESC must once again be hit. The ESC key acts as a toggle switch between keyboard control and arrow key control.

To begin drawing our fault tree, we selected the BUILD option on the menu just brought up by placing the cursor on the word BUILD and pressing the left button on the mouse. This called up the BUILD submenu containing all the constructs needed to create fault tree models (see Figure 18).

NOTE: The left button is used to call up submenus when they exist. The left button also selects function options from menus, and the right button takes down menus.

NOTE: To view the three fault trees (ECS, CCS, and LOSP) rather than rebuilding them:

1. Bring up the Main Graphics menu.
2. Select FILE.
3. Select LIST TREES.
4. Pick one of the fault trees listed.
5. Repeat the process for other fault trees.

Figure 18. The BUILD submenu.

3.3.4 Fill Color

At this point, we could have started drawing our fault tree models but, instead of selecting one of the fault tree construct symbols, we selected the Attributes submenu. This submenu is shown in Figure 19. From here, we chose the FILL COL option to pick a color for our drawing symbols. Choosing the FILL COL option brought up a color selection menu consisting of the sixteen available colors of the EGA monitor (the CGA color selection menu is only four colors long). We placed our cursor on the desired color and pressed the right button on the mouse. The color selection menu disappeared and the attribute line at the top of the screen changed to reflect the new FILL color. In a similar manner, we picked a color for the lines on our fault tree.

Figure 19. Attributes submenu.

3.3.5 Fault Tree Modeling

With these choices made, we are ready to begin drawing. The Attributes submenu was removed by placing the cursor anywhere on the menu and pressing the right button. The first fault tree drawn represented operation of the Emergency Cooling System (ECS).

This system takes water from the storage tank (Figure 8) and injects it into the reactor vessel via two pump trains. Only one of the two pump trains is needed for system success. Since we are interested in system operation while in a loss of offsite power situation, the emergency power supplies for each of the electrical components are shown on the figure.

The first step in our fault tree construction was to define the top event. Because fault trees work in failure space (i.e., what can go wrong), the top event is the occurrence of an undesired event. In this case, the undesired event was "ECS FAILS TO INJECT WATER INTO THE REACTOR VESSEL."

NOTE: Before going any further, it is important to recognize that there are many different ways to model a

system correctly, each method having its advantages and disadvantages. This example fault tree shows a method that points out many of the features of IRRAS 2.0 and may or may not be better than other models that could be made.

We began our fault tree modeling by looking at the system configuration and asking the question, "How can the system fail to inject water into the reactor vessel?" The answer our analyst gave was, "Either the water never gets from the storage tank to the pump trains or both pump trains fail to inject the water into the vessel." Therefore, our top gate is an OR gate with two inputs: (1) no water to the pumps, and (2) both pump trains fail to inject.

To draw this, we selected an OR gate from the BUILD submenu. The cursor disappeared and a ghost OR gate appeared in its place. The ghost could be moved around with the mouse to any place on the screen, even on top of the menus. When we got the OR gate where we wanted it, the left button on the mouse was pressed, the OR gate was set down and another ghost appeared. The right button was pressed and the ghost was replaced with the cursor.

The appearance of another ghost after the first was set in place was caused by having Multi-pick selected as the mode of mouse operation. This mode allows the user to perform a function repeatedly without going back to the menu each time. This feature can be removed by selecting MULTI-PICK on the Attributes submenu. The Attribute header will indicate Single-pick. In the Single-pick mode, the cursor returns after the first execution of a command.

NOTE: A complete display of all the available fault tree symbols and their descriptions are provided in Section 4.1.3.2.

3.3.5.1 Text. To label this gate as our top event, text was added to the block accompanying the OR shape. This was done by selecting the TEXT option from the Main Graphics menu. The TEXT submenu (Figure ?) provides all the commands needed to create titles, labels, descriptions, and names.

Figure 20. Text submenu.

NOTE: With overlay menus, options can be selected from any menu showing as long as the desired option is at least partially visible. Thus, it is sometimes advantageous to bring up submenus as far to the right edge of the parent menu as possible so that options from either menu can be picked.

On the Attribute submenu, the TEXT SIZE option was selected to set the proper size of the text for our present needs. This selection caused the cursor to disappear and the following prompt appeared at the bottom of the screen:

Enter Text Size >

Text sizes are indicated by numbers between 0.01 and 66.00, with 66 being the full 66 lines from the top to the bottom of the screen. This roughly corresponds to the 66 lines on a full sheet of regular paper. For the purpose of writing text in gate blocks,

a text size of about 0.5 is appropriate. This size looks too small on the screen, but is good for sending to a laserjet printer. Larger text sizes will be necessary for printers with lower resolution. So we entered 0.5 using the keyboard and pressed <Enter>. The prompt disappeared, the attribute line updated to show TEXT=0.5, and our cursor returned.

The WRITE TEXT option from the TEXT submenu was picked to begin writing our top event description. The following prompt appeared at the bottom of the screen:

Pick location for description

The cursor was placed in the box at the point about which the first line of text was to be centered and the left button on the mouse was pressed. The cursor went away and a new prompt appeared at the bottom of the screen:

Enter label (terminate with CR) >

The first line of our label, ECS FAILS TO, was entered and <Enter> was pressed. The text appeared at the desired location and the prompt changed to:

Next label (terminate with CR) >

The next line of text, INJECT WATER INTO, was entered <Enter> was pressed, and the second line of text appeared centered under the first line. Once more the "Next label (terminate with CR) >" prompt appeared. The third line of text, THE REACTOR VESSEL, was input and placed on the drawing. Since this was the final line of text to be written, the WRITE TEXT command was terminated by pressing <Enter> without writing any text at the prompt. This gave back the cursor, ready for the next command.

At this point, the top gate had been selected and labeled. To make this gate easy to recognize as the top gate in our fault tree (especially in the alphanumeric logic report), we changed its name from the default name to ECS-TOP. To do this, we went to the TEXT submenu and selected the REP NAME option. REP NAME replaces the current name of an object with a new name supplied by the user. Gate and event names cannot exceed 16 characters.

NOTE: IRRAS 2.0 automatically provides default names for all gates and basic events. Gates are named GATEn, where n is a number beginning with 0. Basic events are given default names consisting of the word EVENT and a number. Transfer gate default names begin with TRANS.

When REP NAME was picked, the following prompt appeared at the bottom of the screen:

Pick entity to be named

We then put the cursor anywhere on the gate symbol and pressed the left button. The cursor disappeared and

Enter name >

showed up on the bottom line. Using the keyboard, we entered the new gate name, ECS-TOP, and pressed <Enter>. The name was written on the gate and the cursor returned.

NOTE: If the entity name was already showing when it was replaced, IRRAS 2.0 will write the new name over the old name. To see the new name clearly, select the SHOW option from the main Graphics menu with the right button on the mouse. This will refresh the screen and redraw the fault tree without the old name present.

At this point, our fault tree looked exactly like Figure 21.

Figure 21. ECS fault tree top event.

NOTE: More on writing text is provided in Section 4.

3.3.5.2 Fault Tree Modeling - Part II. If you recall, the two inputs to our top gate were (1) no water to the pump trains, and (2) both pump trains fail to inject. (If you don't recall, see Section 3.3.5.) There are two ways that water can fail to get to the pump trains. These are (1) failure of the storage tank, and (2) failure of motor-operated isolation valve E-MOV-1 to open. (There are other ways as well, but let's overlook them for simplicity's sake.)

Since either way is sufficient in itself to prevent water from reaching the pump trains, the "no water to the pump trains" input is an OR gate with two inputs. We modeled this by selecting an OR gate from the BUILD submenu and placing it under our top gate. Text was added to this gate, and it was renamed (see Figure 22). One input to this gate, failure of the storage tank, was modeled as a basic event by picking BEVENT from the BUILD submenu and setting the basic event shape down under the OR gate. We named this basic event TANK by using the REP NAME command once again. The other input to the "no water to the pump trains" OR gate is an OR gate labeled E-MOV-1 FAILS TO OPEN.

NOTE: Basic event names are limited to 16 characters. Complemented events (not events) are indicated by preceding the event name with a slash (/). The slash (/) is counted as one of these 16 characters. See Section 6.1 on the treatment of complemented events.

Two ways of having E-MOV-1 fail to open were modeled. One way was to have actual failure of the valve, and the other was to have failure of the power to the valve. Each of these failure modes were modeled as basic events under an OR gate labeled E-MOV-1 FAILS TO OPEN. In the MULTI-PICK mode, the basic event shape was selected and placed using the left button on the mouse. This set the shape in place and, instead of giving back the cursor, another basic event ghost was created. This ghost was positioned and set using the left button. The right button was used to cancel the BEVENT command and give back the cursor.

NOTE: When using the mouse to set basic fault tree constructs (shapes) in the drawing, the MULTI-PICK mode sets the ghost and gives another ghost to set somewhere else. Therefore, you don't need to go back to the BUILD submenu to get a basic shape if you are going to need more than one. The Single-pick mode sets

the ghost and then gives back the cursor.

We named these basic events ECS-MOV-1 and DG-A, representing failure of the motor-operated isolation valve and failure of the Diesel Generator A power supply, respectively. Next, we connected up the inputs to their gates with the LINE command from the BUILD submenu. When LINE was selected with the left button, the following prompt was displayed in the bottom right-hand corner of the screen:

Enter points for line

We moved our cursor to the beginning point of our first line and pressed the left button. We then moved the cursor to where we wanted to end the line, creating a green rubber band line as the cursor was moved. We pressed the left button again and the green line became fixed at both ends and changed to its final color (line color). We continued to draw another line from the last fixed point by moving the cursor to the next desired location and pressing the left button on the mouse. To terminate the line command, we pressed the right button. In the MULTI-PICK mode, after canceling, the prompt returned; we moved the cursor to the starting point for the next line and repeated the process. At this point our drawing looked like Figure 22.

NOTE: When placing shapes or lines, the shape or line will snap to specific locations as an aid to connecting fault tree shapes. This eliminates the need to place the cursor exactly where a shape or line is to be.

Figure 22. Partial ECS fault tree.

3.3.5.3 Saving Our Work. At this time, we decided to break for lunch, even though our fault tree model was not finished (we have our priorities, you know). This meant that we needed to save the work that had been done already. We went to the FILE submenu from the main Graphics menu. This submenu is shown in Figure ?.

Figure 23. File submenu.

The SAVE TREE option was used to save our current fault tree. When SAVE TREE was selected, the cursor disappeared and the following prompt was displayed in the lower left corner of the screen:

Enter file name >

We decided to call our fault tree ECS (probably due to a lack of imagination). Any name up to 16 characters long will work provided the characters are acceptable to the operating system and the first 8 characters are unique. We typed in ECS <Enter> and the prompt disappeared and was replaced by:

Wait ..

As the fault tree information was processed, the number of dots after the "Wait" grew and data were transferred to the hard disk. When the fault tree was completely saved, the prompt went away and the cursor returned. With the fault tree saved and named, we were free to exit the BUILD Fault Trees module and IRRAS 2.0 without fear of losing any of our work.

NOTE: Had we tried to exit the BUILD Fault Trees module without saving the current tree, we would have received a message telling us that the tree was not saved. The message would also state "Enter X to 'EXIT'".

Upon selecting EXIT from either the main Graphics menu or the FILE submenu, the screen went blank and the message

Updating Fault Tree--->ECS

appeared at the bottom of the screen. Then the message

Fault tree successfully loaded

was flashed on the screen. This was subsequently replaced by the Fault Tree Graphics System menu and the message

Data Base Update complete...

From this menu, we exited back to the IRRAS 2.0 main menu and exited out of IRRAS 2.0.

NOTE: Always exit via the menus rather than turning off the machine in the middle of an IRRAS session. This will ensure that your data are not inadvertently lost or corrupted.

3.3.5.4 The Return to Fault Tree Modeling. After a hearty lunch, we were eager to get back to our fault tree (sure!). We entered IRRAS 2.0 as we did the very first time, except this time IRRAS 2.0 remembered all the constants and went directly to the main menu. We went into the BUILD Fault Trees module, verified that we were in the correct family, and selected the BUILD Graphics Trees option. Next, we brought up the main Graphics menu and selected the FILE submenu. Under the FILE submenu we chose LIST TREES. LIST TREES brought up a list of all the fault trees that were extracted from the IRRAS 2.0 data base at the time. (See Section 4.1.2 for more on fault tree extraction.) From this list, we simply put the cursor on ECS and pressed the left button on the mouse. IRRAS 2.0 drew the ECS fault tree on the screen, ready for our use.

Before lunch, we had completed all the inputs to the gate labeled no water to the pump trains, which was one of the inputs to the top event. Next, we began constructing the other input to the top event, both pump trains fail to inject. The simultaneous failure of both pump trains was modeled as an AND gate, with failure of train A and failure of train B as its inputs. To do this, we selected the AND command from the BUILD submenu using the right button on the mouse. The AND gate was placed under the top gate and to the right. We then renamed this gate ECS-TRAINS and labeled it BOTH PUMP TRAINS FAIL TO INJECT.

The inputs to this gate were the failures of the pump trains. We modeled Train A as an OR gate with four basic events, indicating that the train could fail to inject by pump failure, check valve failure, isolation valve failure, or electric power failure. An OR gate was selected from the BUILD submenu and placed under the AND gate. BEVENT was selected next, and four basic events were placed under the OR gate. The LINE command was used to connect the basic events to the OR gate (See Figure 19).

NOTE: When a duplicate event name is input, the computer will ask if you really want to allow a duplicate name.

3.3.5.5 Modeling Symmetry Because Train B is symmetrical to Train A, there was no need to go through the process just described to create Train B. The logic structure of Train B was created instead by copying the Train A logic. We did this by selecting the EDIT submenu from the Graphics menu. This menu is shown in Figure ?.

Figure 24. The Edit submenu.

From this submenu, we selected the COPY command using the left button. This brought up the prompt:

Pick region to be copied

We then moved the cursor to one corner of a box that would entirely surround the shapes to be copied, in this case, the OR gate and four basic events representing a pump train. Pressing the left button on the mouse fixed one corner of this box. Moving the cursor created a rubber band box that stretched with the movement of the cursor. Once the box was drawn where we wanted it, the left button was pressed. This defined the region to be copied. The rubber band box disappeared and the prompt changed to:

Pick reference point

We moved the cursor to the top of the OR gate and pressed the left button. The prompt changed to:

Pick new location

We placed the cursor where we wanted our copied region's reference point to end up on the drawing and pressed the left button. The prompt went away and the computer repeated the shapes within the defined region at the new location. Thus, we quickly created the other pump train and guaranteed that the two identical trains looked identical in the fault tree. The newly created train had default gate and event names. These defaults names are not unique names throughout the fault trees; therefore, the user should rename gates and events. After renaming the gates and basic events (see Figure 25), putting labels on the gates, and connecting all the gates with lines, we added a title to our fault tree using the

WRITE TEXT option after setting the TEXT SIZE to 1.8. Our fault tree looked like Figure 25.

WARNING: If you do not rename all of the gates on a given page and the fault tree is referenced by or references other fault trees, a conflict may occur due to duplicate gate names. A good practice is to change the default gate names.

Figure 25. Completed ECS fault tree.

3.3.5.6 Saving a Previously Saved Drawing. With the ECS fault tree completed, we were ready to begin another fault tree. But first, we had to save all the work done since lunch. Because we saved our work previously using the SAVE TREE option, all work since then was modifying an existing fault tree. To save this work

under the same fault tree name, we used the save tree option (SAVE TREE) on the FILE submenu. When this option was selected, this prompt appeared at the bottom of the screen:

Enter file name or CR for file ECS.DLS >

This command allows the replacement of any fault tree with the current drawing. To have the ECS fault tree updated, we pressed the carriage return. The computer prompts at the bottom of the screen:

File already exists. Do you wish to replace? (Y/N)

Upon selecting Y <ENTER>, the old fault tree is replaced with the modified version. The cursor disappeared, and the computer processed the information and gave the same displays as it did for saving the original drawing. When the work was saved, the cursor returned ready for further use.

3.3.5.7 Saving a Drawing Under More Than One Name. The next step in our problem was to draw the Containment Cooling System (CCS) fault tree. Rather than starting from scratch, we took advantage of the similarities between the ECS and CCS by putting the ECS fault tree under another fault tree name using the SAVE TREE option from the FILE submenu. We did this by selecting SAVE TREE and entering the new fault tree name, CCS, and pressing <Enter>. When this operation was completed, we had two fault trees created, ECS and CCS, that were identical and looked like Figure 25. Since the ECS and CCS were configured identically, the structures of the fault tree models for both systems were the same. Therefore, we only needed to change the basic event names, descriptions, and title to convert the ECS tree to the CCS tree. The CCS fault tree was called up using the LIST TREES option from the FILE submenu. To change the event names, use the EDIT command from the TEXT submenu (see section 4.1.3.9). In order to edit the titles, the titles must be deleted using the DELETE command from the EDIT submenu and then replaced (see section 3.3.5.1). When the required changes were made and saved, the CCS fault tree looked like Figure 26.

NOTE: Version 3.0 of IRRAS will provide an option that will allow the analyst to modify the text using an integrated full screen editor.

Figure 26. CCS fault tree.

3.3.5.8 Modeling Initiating Events. Before continuing on to create accident sequences, we need to create a single event fault tree that represents the accident initiating event. IRRAS 2.0 treats initiating events as top events in the event trees. Therefore, the initiating event must be modeled as a fault tree. In our problem, we derived the initiating event frequency from available data, so our fault tree model was a single basic event as an input to an OR gate. To create this model, we cleared the drawing screen by selecting NEW TREE from the FILE submenu. This brought up a fresh drawing screen.

The simple model of Figure 27 was constructed and saved under the name of LOSEP. This fault tree generated one single-order cut set, LOSEP, the initiating event. Including this fault tree in each accident sequence put the initiating event in every accident sequence cut set.

Figure 27. LOSEP fault tree.

NOTE: The construction of a fault tree consisting of only a basic event would generate an error message stating that IRRAS 2.0 could not find the fault tree top gate. Since an AND gate for a top gate requires at least two inputs to satisfy the logic, an initiating event fault tree of one basic event must have an OR gate as the top gate.

3.3.6 Assignment of Fault Tree Levels

The IRRAS 2.0 data base has the capacity to store a huge number of fault trees. The data base can also handle extremely large fault trees that require many screens to draw. Each screen or page of a fault tree has its own unique identity. This identity is used to call up the graphics and to link the logic to other parts of the fault tree. Listing all the pages of every fault tree when analyzing the fault trees or accident sequences would be cumbersome and confusing. IRRAS 2.0 alleviates this problem by the assignment of each fault tree page to a level. A zero (0) level assignment indicates the presence of an event tree top event gate. Any other assignment indicates that the fault tree is a subtree or a continuation page of a large fault tree. The level assignment does not affect the analysis and processing of the fault trees. However, only zero level fault tree names appear on analysis forms and menus; therefore, subtrees cannot be processed independent of the main fault tree.

NOTE: Designating a fault tree page as a zero level fault tree allows the analyst to call out that tree for processing. Zero level trees can transfer to subtrees or other zero level trees.

To make the level assignments, we exited the graphics mode which returned us to the IRRAS-PC Fault Tree Graphics System menu. Next, we exited back to the IRRAS 2.0 main menu and selected the MODIFY Data Base option. This brought up the Edit Data Base Records menu (Figure 15). Because we wanted to make modifications to fault tree attributes, we selected option 1, Fault Trees, highlighted the DEMO FAMILY (the family containing our fault trees), and pressed <Enter>. The Edit Fault Tree menu, Figure 28, appeared.

Edit Fault Tree

Currently Selected Family —> DEMO FAMILY

Option |M| Exit / Add / Modify / Delete / Locate / Next / Previous
 Batch load Fault Tree logic from file

Fault Tree Number 1
 Fault Tree Name CCS
 Fault Tree Level - (0-root trees, others-sub trees)

Description

5th % -----E----	Median -----E----	Min -----E----	Std. Dev -----E----
95th % -----E----	Mean -----E----	Max -----E----	Skewness -----E----
Min Cut Upper Bound -----E----			Kurtosis -----E----

Figure 28. Edit Fault Tree menu.

We chose the Modify option on this menu by pressing M. The highlight automatically moved to the Fault Tree Name field. Using the <Down-arrow> key, we moved the highlight to the Fault Tree Level field and entered 0, making the CCS tree a zero level fault tree. The highlight automatically moved down to the Description field where we input this description: "Containment Cooling System fault tree." Then we pressed <Enter> and received a

Record Modified

message at the bottom of the screen. At this point, our screen looked like Figure 29.

Edit Fault Tree

Currently Selected Family —> DEMO FAMILY

Option |M| Exit / Add / Modify / Delete / Locate / Next / Previous
 Batch load Fault Tree logic from file

Fault Tree Number 1
 Fault Tree Name CCS
 Fault Tree Level 0 (0-root trees, others-sub trees)

Description Containment Cooling System fault tree

5th % -----E----	Median -----E----	Min -----E----	Std. Dev -----E----
95th % -----E----	Mean -----E----	Max -----E----	Skewness -----E----
Min Cut Upper Bound -----E----			Kurtosis -----E----

Figure 29. Level and description added to the CCS fault tree.

By selecting the NEXT option, the ECS fault tree was called up and the modification process was repeated. The same was done for the LOSP fault tree. When the last fault tree was modified, we pressed E for the EXIT option and hit the <Enter> key. This brought us back to the Edit Data Base Records menu. Exiting once more put us back at the IRRAS 2.0 main menu.

3.4 Assigning Reliability Data

Let's review where we were. We have created fault tree models for the Emergency Cooling System, the Containment Cooling System, and the Loss of Offsite Power initiating event. We also made these fault trees zero level trees, indicating that they are event tree top events.

Our next step was the assignment of reliability data to the fault tree basic events in anticipation of fault tree quantification. For each basic event in the IRRAS 2.0 data base, there is room to store two sets of reliability parameters: the base case and the current case. The base case is input through the MODIFY Data Base module, and the current case is input through the FAULT Tree Analysis or SEQUENCE Analysis module. The base case values are used for the original analysis, and the current case values are used for sensitivity analyses.

NOTE: If only a base case set of reliability values are input, IRRAS 2.0 will set the current case values equal to the base case. If no reliability values are input, IRRAS 2.0 will assign a default value of 1.0 for each reliability parameter. All quantifications are performed using the current case values. Therefore, provisions have been made to let the user set the current case equal to the base case (Section 4.2.1.2) and to update the base case with the current case (Section 4.2.1.1).

The most straightforward means of entering basic event reliability data is event by event through the FAULT Tree Analysis module. Other methods with more versatility are discussed in Sections 4.2.1.1 and 4.2.1.2.

To enter basic event reliability data, we selected the FAULT Tree Analysis option from the IRRAS 2.0 Main Menu. This gave us the IRRAS-PC Fault Tree Analysis System menu shown in Figure 30.

Figure 30. IRRAS-PC Fault Tree Analysis System menu.

From this menu, we selected the Modify Event Data option, which brought up the Set Event Data menu shown in Figure 31.

S e t E v e n t D a t a

Option |E| Exit / Probability / Fault Tree list / Seq. list
Class / Reset to Base Case

← choose Event
← other options

N a m e	D e s c r i p t i o n
C-CV-A	
C-CV-B	
C-MOV-1	
C-MOV-A	
C-MOV-B	
C-PUMP-A	
C-PUMP-B	
DG-A	
DG-B	
E-CV-A	
E-CV-B	
E-MOV-1	

Note: Use <PgUp> or <PgDn> to display more Events.

Figure 31. Set Event Data menu.

This menu listed every basic event contained in the fault trees in the DEMO FAMILY, except for complemented events (not events). A complemented event is automatically given the attributes of the event it complements except it is given a probability of 1 - (probability of the complement event). We added reliability data to the first basic event by selecting option P for Probability, moving the highlight to the first basic event, C-CV-A, and pressing <Enter>. This brought up the Event Data menu shown in Figure 32.

Event Data

Currently Selected Family —> DEMO FAMILY

Event-Attributes	
Event Name	Fail Mode Component Type System Type Component ID
C-CV-A	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Location	Class Attributes
Description	

Base—Uncertainty—Data—Current

L<—Distribution Type—>L

(press Esc for list of types)

1.000E+000<—Value—> 1.000E+000

0<—Correlation Class—> 0

Base—Failure—Data—Current

1<—Calculation Type—>1

(press Esc for list of types)

1.000E+000<— Prob —> 1.000E+000

+0.000E+000<— Lambda—>+0.000E+000

+0.000E+000<— Tau —>+0.000E+000

1.000E+000< Calc Prob> 1.000E+000

Note: Fill in any desired changes in the Current Uncertainty and Failure data for this event and press <Enter>.

Figure 32. Event Data menu.

This menu shows the basic event attributes at the top, the uncertainty data in the bottom left corner, and the failure data in the bottom right corner. Since we did not add any attributes to the basic events through the MODIFY Data Base module, the top of the menu showed only the basic event name. The uncertainty and failure data areas showed the default values of 1.0, entered automatically by IRRAS 2.0. These are the values that we had to change to properly quantify the fault trees.

NOTE: The Event Attributes fields and the base case fields in the Failure Data and Uncertainty Data areas are for information purposes only and are not accessible on this menu. Modifications to these fields must be made through the MODIFY Data Base module.

3.4.1 Failure Probability Calculation Types

When the Event Data menu first appeared, the cursor and highlight were on the first field in the Failure Data area. This

field is the current case Calculation Type. The Calculation Type specifies the method to be used to calculate the basic event failure probability. The various methods available in IRRAS 2.0 were displayed by pressing the <Esc> key. This brought up an overlaid help menu, Failure Data Calculation Types, that listed each calculation type. Our screen looked like Figure 33.

Failure Data Calculation Types	
Type	Calculation Method
1	Probability
2	$\text{Lambda} * \text{Mission Time}$
3	$1 - \text{Exp}(-\text{Lambda} * \text{Mission Time})$
4	$\text{Lambda} * \text{Min}(\text{Mission Time}, \text{Tau})$
5	Operating Component with Repair (Full Eq)
6	$\text{Lambda} * \text{Tau} / 2.0$
7	$1 + (\text{EXP}(-\text{Lambda} * \text{Tau}) - 1.0) / (\text{Lambda} * \text{Tau})$
8	Base Probability + Probability
9	Base Probability * Probability
T	Set to House Event (Failed, Prob=1.0)
F	Set to House Event (Successful, Prob=0.0)

Event Name C-CV-A	ID
Location Class Description	15 16

Base—Uncertain	nt
L<—Distribu	>1
(press Esc for	
1.000E+000<—Va	
0<—Correlat	

Calculation Type —>1	00
+0.000E+000<— Tau —>+0.000E+000	00
1.000E+000< Calc Prob> 1.000E+000	

Note: Fill in any desired changes in the Current Uncertainty and Failure data for this event and press <Enter>.

Figure 33. Failure Data Calculation Types help menu.

A description of each calculation type follows.

Calculation Type 1 takes the number specified by the user in the Probability field as the basic event failure probability.

Calculation Type 2 uses the number provided as Lambda as the basic event failure rate per hour and multiplies it by the system mission time expressed in hours. The mission time is set by the user in the UTILITY Options module (Define Constants) or the FAULT Tree Analysis module. A default mission time of 24 hours is provided by IRRAS 2.0 until it is changed by the user. This probability calculation is a rare event approximation to the actual failure probability for an operating component without repair during the mission time. This approximation is relatively good for failure probabilities less than 0.1.

Calculation Type 3 uses the actual equation for failure probability for an operating component without repair,

$$P = 1 - \text{EXP}(-\lambda t) \quad (1)$$

where

P = failure probability of the basic event,

λ = failure rate per hour, input as Lambda, and

t = mission time expressed in hours, input as a default.

Calculation Type 4 is a rare event approximation for the failure of an operating component with repair. The approximation is Lambda times Tau. It uses Lambda as the per hour failure rate and Tau as a user-specified time to repair. If the mission time is less than Tau, then Lambda*Tau is a better approximation of the event solution; therefore, IRRAS 2.0 uses Lambda times the minimum of Tau and mission time.

Calculation Type 5 is the actual equation for the failure probability of an operating component with repair. The equation is

$$P = [\lambda T / (1 + \lambda T)] \{ 1 - \text{EXP}[-(\lambda + 1/T)t] \} \quad (2)$$

where

P = failure probability of the basic event,

λ = failure rate per hour, input as Lambda,

t = mission time expressed in hours, input as a default, and

T = average time to repair expressed in hours, input as Tau.

Calculation Type 6 is the rare event approximation for the failure probability of a standby component with a surveillance test interval. The equation used is

$$P = \lambda * T / 2 \quad (3)$$

where

P = failure probability of the basic event,

λ = standby failure rate per hour, input as Lambda, and

T = surveillance test interval in hours, input as Tau.

Calculation Type 7 is the actual equation for the failure probability of a standby component with a surveillance test interval. The equation is

$$P = 1 + [\text{EXP}(-\lambda T) - 1] / \lambda T \quad (4)$$

where

P = failure probability of the basic event,

λ = standby failure rate per hour, input as Lambda, and

T = surveillance test interval in hours, input as Tau.

Calculation Types 8 through 11 are used for sensitivity analyses. Type 8 allows the user to specify a current case probability that differs from the base case by an exact amount. The amount to change the base case probability by is entered in the probability field. Type 9 lets the user create a current case probability that is a specified percentage of the base case. The percentage is entered in the probability field. Calculation Type 10 turns the basic event into a house event that is always true (probability 1.0), and Type 11 turns the basic event into a house event that is always false (probability 0.0).

Setting an event to a house event actually changes the logic of the fault tree, pruning appropriate branches and events from the fault tree. Therefore, the flags on the affected fault trees will indicate a need to generate new cut sets rather than just requantifying existing cut sets. See Section 6.1 for details on the processing of house events.

For our example problem, the data base provided in Table 1 contained the failure probability for check valves, the component of concern for basic event C-CV-A. Therefore, we chose Calculation Type 1 by putting the highlight on that option and pressing <Enter>. The help screen disappeared, a 1 was placed in the current case Calculation Type field, and the cursor and highlight returned to this field.

NOTE: Once you have become familiar with IRRAS 2.0, you will be able to enter the proper calculation type without the use of the help menu. To do so, just type in the number for the desired calculation type in the Calculation Type field. For the purpose of this DEMO, all Calculation Types are 1.

Next, we moved the cursor to the Prob field and entered the failure probability for check valves as listed in Table 1: 1.0E-4.

IRRAS 2.0 will accept the input as short-hand scientific notation (as shown) or decimal input (0.0001).

NOTE: When using the short-hand scientific notation, a decimal point must precede the "E", thus 1E-2 will not be accepted but 1.E-2 or 1.0E-2 will. IRRAS 2.0 will accept an upper-case E or a lower-case e. Also note that 1.0E-020 is not the same as 1.0E-02. This has caused confusion in the past.

3.4.2 Uncertainty Analysis Data

From the Failure Data area, we moved to the Uncertainty Data area using the arrow keys or the tab key. The fields in this area that can be accessed from this menu are the current case Distribution Type, a distribution parameter Value, and a Correlation Class.

Currently, IRRAS 2.0 supports normal, lognormal, beta, gamma, uniform, and exponential distributions for the Monte Carlo uncertainty analyses. A help screen has been created for the Distribution Type field similar to the one for Calculation Type. It is also accessed by pressing the <Esc> key. The default distribution type is the lognormal. We put the cursor on the Value field and entered 3.0 as the 95% error factor for the desired lognormal distribution.

NOTE: Most distributions can be defined with two statistical parameters. For the lognormal distribution, IRRAS 2.0 uses the mean failure probability and the 95% error factor to define the distribution. The mean failure probability is calculated from the data input in the Failure Data area just discussed. The 95% error factor, which is the ratio of the 95th percentile value to the median value for the distribution, is supplied by the user in the Value field in the Uncertainty Data area. The other distributions are discussed in section 5.4.

After entering the error factor, we moved the highlight to the Correlation Class field and input a correlation class number. Correlation class numbers are used by IRRAS 2.0 to account for data dependencies among like events in the data base. Correlation class numbers can range from 0 to 9999. A zero (0) correlation class number indicates that there are no data dependencies. Like events, such as motor-operated valves failing to open on demand, should be assigned the same correlation class number. IRRAS 2.0 will then use the same sample value for all basic events with that correlation class number. This provides a more accurate treatment of the uncertainty. At this point, the screen looked like Figure 34.

Event Data

Currently Selected Family —> DEMO FAMILY

Event-Attributes																					
Event Name	Fail Mode	Component	Type	System Type	Component ID																
C-CV-A						1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Location	Class Attributes																				
Description																					

Base—Uncertainty—Data—Current

L<—Distribution Type—>L

(press Esc for list of types)

1.000E+000<—Value—> 3.000E+000

0<—Correlation Class—> 1

Base—Failure—Data—Current

1<—Calculation Type—>1

(press Esc for list of types)

1.000E+000<— Prob —> 1.000E-004

+0.000E+000<— Lambda—>+0.000E+000

+0.000E+000<— Tau —>+0.000E+000

1.000E+000< Calc Prob> 1.000E+000

Note: Fill in any desired changes in the Current Uncertainty and Failure data for this event and press <Enter>.

Figure 34. Completed Event Data menu for basic event C-CV-A.

NOTE: The user must set up a correlation class numbering scheme for the basic events in the data base. For example, correlation class 1 may be assigned to motor-driven pumps fail to start, correlation class 2 to motor-driven pumps fail to continue to run, correlation class 3 to check valves fail to close, and so on. Currently, this scheme is not saved within IRRAS 2.0 but may be included in the future.

Once all fields on the menu were set as desired, we pressed the <Enter> key to send the information to the data base. IRRAS 2.0 gave us the message

Requested changes made to the event

and put us back at the Set Event Data menu shown in Figure 31. From here, we selected the next basic event and repeated the entire process. We did this until we had assigned failure and uncertainty data to every basic event in the data base. Refer to Table 1 for the values entered.

NOTE: The data entered in the fashion just described become the current case data. The base case data remain at the default values of 1.0 until changed in the MODIFY Data Base module. Selecting the Reset to Base Case option on the Set Event Data menu (Figure 31) before updating the base case will cause the data entered to be deleted. See Sections 4.2.1.1 and 4.2.1.2 for information on updating the base case data.

IRRAS 2.0 provides more sophisticated ways of entering failure and uncertainty data that reduce the amount of data input required and ensure consistency among like basic events. These techniques are discussed in Section 4.2.1.2.

3.5 Analyzing Fault Trees

After creating the fault tree models and assigning failure and uncertainty data to the basic events, we were ready to generate the fault tree minimal cut sets, quantify the fault tree top events, and perform the top event uncertainty analysis. All of these functions are done in the FAULT Tree Analysis (select FAULT on IRRAS 2.0 Main menu) module from the Analyze Fault Trees option, A, on the IRRAS-PC Fault Tree Analysis System menu.

Selecting the Analyze Fault Trees option brought up the Analyze Fault Trees menu shown in Figure 35. This menu provided a list of the zero level fault trees in the DEMO FAMILY along with their descriptions. To the left of the tree names were sets of characters and an arrow. The presence or absence of these characters indicated the status of our analysis. The meaning of each character is :

- c - The cut sets have not been generated or the fault tree logic or basic event data have changed since the cut sets were generated.
- q - The cut sets have not been quantified or the cut sets or basic event data have changed since the cut sets were last quantified.
- u - The uncertainty analysis has not been performed or the cut sets or basic event data have changed since the uncertainty analysis was performed.
- f - The fault tree logic has been modified without updating the graphics.

The above flags only affect the zero level fault trees. If any changes are made to a zero level fault tree, the flags will indicate the need for quantification. If any changes are made to any other level of fault trees, the flags will not indicate a need for quantification.

Pressing the <ESC> key when on the menu in Figure 35 will bring up a help menu with these definitions. Pressing <Enter> causes the help screen to disappear.

Figure 35. Analyze Fault Trees menu.

As you can see in Figure 35, each of our trees had the "c" indicator or flag meaning that the cut set generation has not been performed.

NOTE: The DEMO data base provided has had all the calculations performed; therefore, the flags present will not be the same as those discussed in the example.

3.5.1 Generating Minimal Cut Sets and Quantifying the Fault Tree

We began our analysis by selecting C for cut set generation and highlighting the CCS fault tree. IRRAS 2.0 then gave the message

Reading in basic events - please stand by

All the Basic event data is read into memory. After the failure data are read into memory, the Cut Set Generation Cutoff Values screen appeared. This screen is shown in Figure 36.

Analyze Fault Trees

Currently Selected Family —> DEMO FAMILY

Option |C| Exit / Cut Set Generation / Quantification / Uncertainty calc.

———— CUT SET GENERATION CUTOFF VALUES ————

Perform Probability Cutoff? (Y/N) Y	Cutoff Value	1.000E-015
Perform Cut Set Size Cutoff? (Y/N) Y	Size Cutoff	6
Mission Time 2.400E+001		

Figure 36. Cut Set Generation Cutoff Values menu.

This menu was used to select our cutoff criteria for the generation of the fault tree minimal cut sets. Cutoff criteria were used to limit the number of cut sets generated to a manageable number and to speed up the processing time. For extremely large or complex fault trees, cutoff criteria may be required to allow IRRAS 2.0 to process the tree.

The probability cutoff tells IRRAS 2.0 to not generate any cut sets with a probability less than the specified value. The size cutoff tells IRRAS 2.0 to not generate any cut sets with more basic events than the specified value. The defaults for these criteria are probability cutoff in effect with a minimum cut set probability of 1.0E-15 and size cutoff in effect with a maximum cut set size of 6 basic events per cut set. These defaults can be permanently changed in the UTILITY Options module (Define Constants). Changes for just the selected tree can be made on this menu.

Also on this menu is the mission time for the quantification. This mission time is the same as that set in the defaults on the IRRAS Constants menu. Changing the mission time in the FAULT Tree Analysis module (or SEQUENCE Analysis module) does not permanently change the mission time. As soon as the IRRAS 2.0 system is exited, the mission time will revert back to the default set in the IRRAS 2.0 Constants menu.

We chose not to change the defaults for our analysis, so we pressed <Enter>. If Cut Set Generation was selected without highlighting a fault tree, a warning message appears. The screen would look like Figure 37.

A n a l y z e F a u l t T r e e s

Currently Selected Family —> DEMO FAMILY

Option |C| Exit / Cut Set Generation / Quantification / Uncertainty calc.

— FAULT TREES —

All fault trees will be updated as specified since no individual fault tree was highlighted. This may take considerable time depending upon the number of fault trees required to be updated as indicated by the flags preceding the fault tree names and the kind of updates required.

Do you wish to continue? |N|

f-Fault Trees, c-Cut Sets, q-Quantification, u-Uncertainty
More change information may be obtained for each tree using <Esc>.
Use <PgUp> and <PgDn> to display more fault trees.

Figure 37. Batch Fault Tree Processing warning message.

NOTE: Selecting a fault tree analysis option without highlighting a specific fault tree will cause IRRAS 2.0 to process all fault trees needing the specified type of analysis. IRRAS 2.0 will then provide a warning and allow the user to verify his intent. If the batch operation was inadvertently selected, the user can return to the previous menu by indicating no desire to continue.

If Cut Set Generation was selected and a fault tree highlighted, IRRAS 2.0 would give the following message:

Reading fault tree from data base

and then begin processing the CCS fault tree. IRRAS 2.0 keeps us informed of the status of the analysis at each step. It indicates which gate is being processed, the number of intermediate cut sets that have been generated, when it was expanding the gates, when it was separating out independent subtrees, when it was coalescing gates, when it was supersetting the cut sets, and when it was sorting the cut sets. When all this is completed, IRRAS 2.0 provides a summary of the cut set generation and quantification. For our analysis of the CCS fault tree, the summary looked like Figure 38.

Analyze Fault Trees			
Currently Selected Family —> DEMO FAMILY			
Option C Exit / Cut Set Generation / Quantification / Uncertainty calc.			
Fault Tree	Elapsed Time		
CCS	00:00:03.020		
Cut Set Size	UpperBound		
1	3	2.098E-002	
2	12	3.161E-004	
3	0	+0.000E+000	
4	0	+0.000E+000	
5	0	+0.000E+000	
6	0	+0.000E+000	
7	0	+0.000E+000	
8	0	+0.000E+000	
9	0	+0.000E+000	
10	0	+0.000E+000	
>10	0	+0.000E+000	
Total	15	2.129E-002	
Press <Ctrl-A> to Abort			

Figure 38. CCS fault tree cut set generation summary.

The summary showed that the CCS fault tree generated three single failure cut sets and 12 doublets for a total of 15 minimal cut sets. The point estimate calculation of the minimal cut set upper bound was 2.13E-2, and the total time required for the processing was 3.02 seconds. More details of the analysis are available in the Display Fault Trees option in the FAULT Tree Analysis module and in the REPORT on Data Base module.

IRRAS 2.0 provided a message stating

Press Enter Key to continue

when it put up the summary of the analysis. So when we were finished with the summary, we pressed <Enter>. IRRAS 2.0 indicated that the cut set generation was completed and returned us to the Analyze Fault Tree menu. The status next to the CCS fault tree was updated to show only the uncertainty analysis remaining to be completed. This is shown in Figure 39.

Analyze Fault Trees

Currently Selected Family —> DEMO FAMILY

Option |C| Exit / Cut Set Generation / Quantification / Uncertainty calc.

	FAULT	TREES
u--> CCS		Containment Cooling system fault tree
c --> ECS		Emergency Cooling system fault tree
c --> LOSP		Loss of Offsite Power Initiating Event fault tree

Note : Changed indicators precede each fault tree with meanings as follows-
f-Fault Trees, c-Cut Sets, q-Quantification, u-Uncertainty
More change information may be obtained for each tree using <Esc>.
Use <PgUp> and <PgDn> to display more fault trees.

Figure 39. Analyze Fault Trees menu after CCS cut set generation.

The cut sets for the remaining fault trees were also generated as just explained.

3.5.2 Performing Fault Tree Uncertainty Analyses

After generating the minimal cut sets, we were ready to perform the uncertainty analyses. We started by selecting U for the Uncertainty analysis option on the Analyze Fault Trees menu, highlighting the CCS fault tree, and pressing <Enter>. The Uncertainty Calculation Values menu was brought up as shown in Figure 40.

Analyze Fault Trees

Currently Selected Family —> DEMO FAMILY

Option |U| Exit / Cut Set Generation / Quantification / Uncertainty calc.

—UNCERTAINTY CALCULATION VALUES—

Number of samples to use in Monte Carlo simulation	1000
Seed for random number generator	0

Note: Use "0" as the seed to get a random seed from the clock.

Change to desired uncertainty values and press Enter

Figure 40. Uncertainty Calculation Values menu.

On this menu, we set the desired number of samples for the uncertainty analysis and the seed value for the random number generator. Note, if "0" is used as the seed, the IRRAS program generates a random seed. Figure 40 shows the default values set into IRRAS 2.0. These defaults can be permanently changed in the UTILITY Options module (Define Constants) or changed for just this fault tree on this menu. Since we were satisfied with the defaults, we pressed <Enter>. IRRAS 2.0 flashed the message

Reading in basic events - please stand by

IRRAS 2.0 began the Monte Carlo analysis and provided the status along the way. The screen showed the name of the fault tree being processed, the total number of samples chosen, and the current sample being calculated. For each sample, IRRAS 2.0 shows the value calculated for the mean value of the distribution. Upon completion of the uncertainty analysis, IRRAS 2.0 displayed a summary of the results. The summary for the CCS analysis is shown in Figure 41.

Figure 41. Uncertainty results summary for the CCS fault tree.

The summary provided the boilerplate information such as the fault tree name, the number of minimal cut sets for the top event, the number of basic events in the fault tree, the random seed used, and the number of samples calculated. The summary also gave the results of the calculations including the point estimate calculation; the mean of the distribution for the top event; the 5th percentile, median, and 95th percentile values of the distribution; the minimum and maximum sample values; and the various statistical moments of the distribution.

When we had finished reviewing the summary results, we pressed <Enter>. IRRAS 2.0 indicated that the uncertainty analysis was completed and the Analyze Fault Trees menu returned. We performed the uncertainty analyses for the remaining fault trees using the method just described.

The Uncertainty calc. option has the ability to perform batch analyses on all the fault trees that are not up-to-date. This works just like the batch mode for cut set generation. Select the Uncertainty analysis option without highlighting a fault tree and answer yes <Y> to the desire to continue question. After setting the desired Uncertainty Calculation Values, IRRAS 2.0 will then process one tree after another using the specified random seed and number of samples for each tree.

3.5.3 Displaying the Fault Tree Analysis Results

At this point, we had completed the creation of the fault tree logic models, generated the minimal cut sets, quantified the models, and performed uncertainty analyses. We next desired to look at our results. (Actually we looked at our results at every step of the analysis, which seems to be a good practice, but it is more convenient to discuss how we did it now.)

To look at our results, we exited from the Analyze Fault Trees menu back to the IRRAS-PC Fault Tree Analysis System menu. From this menu, we chose the Display Fault Trees option by pressing D <Enter>. This brought up the Display Fault Tree Data menu shown in Figure 42.

Display Fault Tree Data

Currently Selected Family —> DEMO FAMILY

Option [E] Exit / Cut Sets / Importance measures / Uncertainty data

—FAULT TREES—

CCS	Containment Cooling system fault tree
ECS	Emergency Cooling system fault tree
LOSP	Loss of Offsite Power Initiating Event fault tree

Note : Use <PgUp> and <PgDn> to display more fault trees.

Figure 42. Display Fault Tree Data menu.

From this menu, we were able to look at the analyses results in more detail. Options on this menu provided us with the ability to look at the fault tree cut sets and quantification, the uncertainty analysis results, and some importance measures for the basic events.

To review the fault tree cut sets, we chose the Cut sets option by pressing C, highlighting the desired fault tree, and pressing <Enter>. IRRAS 2.0 displayed

Loading event data

and then brought up the Fault Tree Cut Sets display shown in Figure 43.

FAULT TREE CUT SETS						
Family Name DEMO FAMILY			Fault Tree Name CCS			
Min Cut	2.120E-00	Num	15	Partition ==>	2.120E-002	100.00% Num 15
Option B Exit/ Partition/ Report/ Basic Events <= choose a cut set below						
Num	%	Frequency	Event Names			
1	94.33	2.000E-002	DG-B			
2	4.72	1.000E-003	C-MOV-1			
3	0.47	1.000E-004	DG-A	C-MOV-B		
4	0.28	6.000E-005	DG-A	C-PUMP-B		
5	0.12	2.500E-005	C-MOV-A	C-MOV-B		
6	0.07	1.500E-005	C-MOV-A	C-PUMP-B		
7	0.07	1.500E-005	C-PUMP-A	C-MOV-B		
8	0.04	9.000E-006	C-PUMP-A	C-PUMP-B		
9	0.01	2.000E-006	DG-A	C-CV-B		
10	0.00	5.000E-007	C-CV-A	C-MOV-B		
11	0.00	5.000E-007	C-MOV-A	C-CV-B		
12	0.00	3.000E-007	C-CV-A	C-PUMP-B		
Use <PgUp> or <PgDn> to display more Cut Sets						

Figure 43. Fault Tree Cut Sets display.

The top of this display gave the family and fault tree names, the minimal cut set upper bound value as calculated by the point estimate quantification, along with the quantity of designated subset items as indicated by the partition number. The lower portion showed the quantified minimal cut sets. The first column gave the ranking of the cut sets according to probability. The second column indicated an approximate percentage contribution of the cut set to the minimal cut set upper bound. The third column gave the frequency of each individual cut set, and the last column gave a listing of the basic events making up each cut set. More cut sets were reviewed by using the <PgUp> and <PgDn> keys.

We were able to perform some additional analysis on the cut sets by using the partition option. After selecting the "Partition" option, the screen in Figure 44 was displayed. This option allowed us to partition the cut sets by selecting only those cut sets which contained a qualified event.

Partition Cut Sets

Currently Selected Family —> DEMO FAMILY
 Total Family Events —> 18
 Total Qualified Events —> 0

Option |1| Exit / Include / eXclude / Compliment / Reset / View Events

Event Name	Fail Mode	Component Type	System Type	Component ID
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16				
Location	Class Attributes			

Note: Enter the attributes of the events to be included. Fields which are left blank are not used. The ? may be used for positional "don't care" characters. The * may be used for non-positional variable length "don't care" strings.
 e.g. AB?CD* matches ABXCD through ABXCDXXXXXXXXXX

Figure 44. Partition Cut Sets display.

Events could be qualified in groups by Including or eXcluding them or individually by selecting the View Events option. In this example, we selected to include all events which contained the string MOV somewhere in the name. The screen in Figure 45 was displayed.

Partition Cut Sets

Currently Selected Family —> DEMO FAMILY
 Total Family Events —> 18
 Total Qualified Events —> 6

Option |1| Exit / Include / eXclude / Compliment / Reset / View Events

Event Name	Fail Mode	Component Type	System Type	Component ID
MOV				
		1	2	3 4 5 6 7 8 9 10 11 12 13 14 15 16
Location	Class Attributes			

Note: Enter the attributes of the events to be included. Fields which are left blank are not used. The ? may be used for positional "don't care" characters. The * may be used for non-positional variable length "don't care" strings.
e.g. AB?CD* matches ABXCD through ABXCDXXXXXXXXXX

Figure 45. Partition included events display.

We could have specified other attributes to further qualify an event or used eXclude to remove some of the qualified events from the list. We selected View Events to display the basic events in this family. The screen in Figure 46 was displayed.

View Events

Option |I| Exit / Include event <← choose Event

Event Name	Event Description
C-CV-A	CCS Train A pump discharge check valve
C-CV-B	CCS Train B pump discharge check valve
+ C-MOV-1	CCS suction isolation valve
+ C-MOV-A	CCS Train A pump discharge isolation valve
+ C-MOV-B	CCS Train B pump discharge isolation valve
C-PUMP-A	CCS Train A motor-driven pump
C-PUMP-B	CCS Train B motor-driven pump
DG-A	Emergency diesel generator A
DG-B	Emergency diesel generator B
E-CV-A	ECS Train A pump discharge check valve
E-CV-B	ECS Train B pump discharge check valve
+ E-MOV-1	ECS suction isolation valve

Note: Use <PgUp> or <PgDn> to display more Events. Included events are indicated by "+".

Figure 46. View Events display.

The currently qualified events were displayed with a "+" in front of the name. At this time, we could have individually toggled a basic event from qualified to not qualified by highlighting the event and pressing the enter key. We selected the Exit option to return to the Partition menu and then selected the Exit option once again to get back to the Cut Set display menu. The screen in Figure 47 was displayed.

Figure 47. Partitioned Cut Set display.

Notice that only those cut sets including an event whose name contains the string MOV were displayed. Also note that the top of the display now contains different values for the current partition contributions.

We decided to get a hard copy report of the partitioned cut sets, so we selected the Report option. This option generated the report shown in Figure ?.

Partition Cut Set Report					
Family Name ->DEMO FAMILY			Fault Tree Name ->CCS		
Mincut Upper Bound 2.120E-002			This Partition 1.156E-003		
Cut No.	% Total	% Cut Set	Freq.	Cut Sets	
1	4.7	4.7	1.0E-003	C-MOV-1	
2	5.2	.5	1.0E-004	DG-A, C-MOV-B	
3	5.3	.1	2.5E-005	C-MOV-A, C-MOV-B	
4	5.4	.1	1.5E-005	C-PUMP-A, C-MOV-B	
5	5.4	.1	1.5E-005	C-MOV-A, C-PUMP-B	
6	5.4	.0	5.0E-007	C-MOV-A, C-CV-B	
7	5.5	.0	5.0E-007	C-CV-A, C-MOV-B	
8	5.5	.0	9.0E-006	C-PUMP-A, C-PUMP-B	
9	5.5	.0	2.0E-006	DG-A, C-CV-B	
10	5.5	.0	5.0E-007	C-CV-A, C-MOV-B	
11	5.5	.0	5.0E-007	C-MOV-A, C-CV-B	
12	5.5	.0	3.0E-007	C-CV-A, C-PUMP-B	
13	5.5	.0	3.0E-007	C-PUMP-A, C-CV-B	
14	5.5	.0	1.0E-007	TANK	
15	5.5	.0	1.0E-008	C-CV-A, C-CV-B	

Figure 48. Partitioned Cut Set report.

More information on specific cut sets was available by choosing the Basic event option, highlighting one of the cut sets, and pressing <Enter>. This called up the Cut Set Basic Event Descriptions display. Figure 49 shows this display.

CUT SET BASIC EVENT DESCRIPTIONS						
Family DEMO FAMILY		Fault Tree CCS			Min Cut 2.120E-002	

Event Name	Description					
	Probability	Comp ID	System	Type	FM	Location
DG-B	Emergency diesel generator B 2.000E-002	EP-DG-B	EP	DG	C1	FZ7

Use <PgUp> or <PgDn> to display more Events
Press <Enter> to Return

Figure 49. Fault tree Cut Set Basic Event Descriptions display.

This display provided all the available information on each basic event contained in the chosen cut set. In our example, the only additional information was the basic event probability. Had we given the basic events descriptions and classified them according to system, component, failure mode and location, this information would have also been displayed. The assignment and use of these types of basic event attributes is discussed in Section 4.2.1.1.

We returned to the Display Fault Tree Data menu by pressing <Enter>, selecting E for Exit, and pressing <Enter> again. From here, we were able to look at the uncertainty results for any and all fault trees in the family by selecting the Uncertainty data option, U, highlighting the desired fault tree, and pressing <Enter>. This brought up the Uncertainty Data display shown in Figure 50.

UNCERTAINTY DATA

Family DEMO FAMILY
 Fault Tree CCS
 Description Containment Cooling System fault tree

Mean	2.061E-002	5th %	1.542E-003	Std. Dev	3.776E-002
Mincut	2.120E-002	Median	9.407E-003	Minimum	3.669E-004
		95th %	7.255E-002	Maximum	3.933E-001

Press <Enter> to return to previous form.

Figure 50. Fault tree Uncertainty Data display

This display gave us the most important results from the uncertainty analysis. Pressing <Enter> returned us to the Display Fault Tree Data menu.

The last display of the analysis results we used was the basic events importance measures. This option was selected by pressing I, highlighting the desired fault tree, and pressing <Enter>. IRRAS 2.0 flashed several messages on the screen that reported the status of loading the cut sets, calculating the importance measures, and sorting the basic events. Then IRRAS 2.0 put the Basic Event Importance display on the screen. The screen looked like Figure 51.

Figure 51. Fault tree Basic Event Importance display.

This display showed every basic event in the chosen fault tree. For each basic event, IRRAS 2.0 calculated and displayed three importance measures. These importance measures are:

- o Fussell-Vesely Importance, an indication of the percentage of the minimal cut set upper bound contributed by the basic event.
- o Risk Reduction Ratio, an indication of how much the minimal cut set upper bound would decrease if the basic event were made perfect (never fails).
- o Risk Increase Ratio, an indication of how much the minimal cut set upper bound would go up if the basic event were always failed.

The Basic Event Importance display listed the basic event name, number of occurrences, its failure probability, and the three importance measures. The basic events were listed in order of Fussell-Vesely importance, high to low. These events can also be sorted by name, probability, reduction, and increase. The Fussell-Vesely Importance, Risk Reduction Ratio, and Risk Increase Ratio came about from choosing ratio on the IRRAS Constants menu

(Figure 10) when the question was asked, Use Ratios in Importance Measures?. If interval had been chosen, the values shown would be the Birnbaum importance measures, risk reduction interval, and risk increase interval. For a detailed discussion of importance measures, see Section 6.5.

3.5.4 Editing the Cut Sets

Once you have generated the minimal cut sets, you may find it necessary to make some minor, or even major, changes. To make these changes you could go back to the Build menu, make your changes to the tree or sequence, and regenerate the cut sets. However, the Cut Set Editor will allow you to make essential changes directly to the cut sets without effecting the sequence or fault tree.

From the Fault Tree Analysis System menu, we selected the Cut Set Editor option, which brought up the a Cut Set Editor menu, shown in Figure 52. Here we positioned the cursor over the CCS cut set and hit enter. (Since "S" is the default menu option, we directly selected our cut set.) This brought up the menu for the Cut Set Editor and displayed our cut sets, shown in Figure 53.

Cut Set Editor

Currently Selected Family —> DEMO FAMILY

Option |S| Exit / Select a Fault Tree Cut Set

CCS	Containment Cooling System fault tree
ECS	Emergency Cooling system fault tree
LOSP	Loss of Offsite Power initiating event fault tree

NOTE: To select a Fault Tree, enter option <S>, move the cursor to the desired Fault Tree, press <Enter>. Use <PgUp> or <PgDn> to display more trees.

Figure 52. Select a Cut menu.

C U T S E T E D I T O R			
Exit / Add / Modify / Delete / Locate / Next / Previous Insert / Replace / Search / Options / Undo Delete / Esc for more help			
Set #	Event Names		
1	DG-B		
2	TANK		
3	C-MOV-1		
4	C-MOV-A	C-MOV-B	
5	C-MOV-A	C-CV-B	
6	C-MOV-A	C-PUMP-B	
7	DG-A	C-MOV-B	
8	DG-A	C-CV-B	
9	DG-A	C-PUMP-B	
10	C-PUMP-A	C-MOV-B	
11	C-PUMP-A	C-CV-B	
12	C-PUMP-A	C-PUMP-B	
13	C-CV-A	C-MOV-B	
14	C-CV-A	C-CV-B	
15	C-CV-A	C-PUMP-B	

Figure 53. Cut Set Editor menu and window.

The Cut Set Editor has 13 menu options, each of which requires only a single key stroke to initiate each command:

- o Exit. End your editing session.
- o Add. Allow addition of an event into a cut set or the addition of an entire cut set.
- o Modify. Permit the user to make a change to a given event name or to position the cut set differently within the data.
- o Delete. Remove an event from within a cut set or remove the entire cut set.
- o Locate. Find the first occurrence of a given string.
- o Next. Find the next occurrence of located string.
- o Previous. Find the previous occurrence of the located string.

- o Insert. Add a given event name to a range of cut sets containing a given event name search string.
- o Replace. Replace a given event name search string with a given event name, within a range.
- o Search. Look for the first occurrence, beginning at the top of the current page, of a given event name search string.
- o Options. Set the range defaults for the Cut Set Editor.
- o Undo Delete. Reverse the results of a deletion, which has just been made.
- o Esc. Toggle between the command menu and the cursor help menu.

We first hit the <Esc> key and noticed that the command menu was replaced by the cursor help menu, as shown in Figure 54. Here we saw that to move the cursor about the editing window we used the keyboard cursor pad. The arrow keys moved us one field in each respective direction. The Home key places the cursor in the upper right corner of the window. And the End key moves the cursor to the bottom right. The PgUp key moved the window up one page (15 lines). PgDn moved the window down one page. Ctrl-Home, and Ctrl-End moved us to the top and bottom of the file.

CUT SET EDITOR				
< >Cursor Up < >Cursor Down < >Cursor Left < >Cursor Right <End>End of Page <Home>Top of Page <PgUp>Page Up <PgDn>Page Down <Ctrl-PgUp>Top of File <Ctrl-PgDn>End of File <Esc> for more				
Set #	Event Names			
1	DG-B			
2	TANK			
3	C-MOV-1			
4	C-MOV-A	C-MOV-B		
5	C-MOV-A	C-CV-B		
6	C-MOV-A	C-PUMP-B		
7	DG-A	C-MOV-B		
8	DG-A	C-CV-B		
9	DG-A	C-PUMP-B		
10	C-PUMP-A	C-MOV-B		
11	C-PUMP-A	C-CV-B		
12	C-PUMP-A	C-PUMP-B		
13	C-CV-A	C-MOV-B		
14	C-CV-A	C-CV-B		
15	C-CV-A	C-PUMP-B		

Figure 54. Cut Set Editor cursor help menu.

After becoming familiar with the cursor movements, we hit the <Esc> key again to show the command menu once more. Next we illustrated the functions of each of the available commands.

We first moved the cursor into the first column and row, or the number column, and typed A to invoke the Add command. The Add message appeared just below the title bar. With the cursor in the number column, the Add command understood that we wished to insert an entire cut set. The window was cleared and the cursor was placed in the first row of the second column, or the first event name column as shown in Figure 55. Then we keyed in the event names, (MYEVENT, YOUREVENT, HISEVENT), of our new cut set, using the Tab key to move between fields and the Enter key to end the addition. The window was then updated with the new cut set in the first row, with the previous data moved down one row.

C U T S E T E D I T O R

Add
 Enter the Event name to be added to this Cut Set
 Use <Enter> when done, and <Esc> to abort

Set #	Event Names	YOUREVENT	HISEVENT	
1	-			
2	DG-B			
3	TANK			
4	C-MOV-1			
5	C-MOV-A	C-MOV-B		
6	C-MOV-A	C-CV-B		
7	C-MOV-A	C-PUMP-B		
8	DG-A	C-MOV-B		
9	DG-A	C-CV-B		
10	DG-A	C-PUMP-B		
11	C-PUMP-A	C-MOV-B		
12	C-PUMP-A	C-CV-B		
13	C-PUMP-A	C-PUMP-B		
14	C-CV-A	C-MOV-B		
15	C-CV-A	C-CV-B		

Figure 56. Event addition message.

CUT SET EDITOR				
Exit / Add / Modify / Delete / Locate / Next / Previous Insert / Replace / Search / Options / Undo Delete / Esc for more help				
Set #	Event Names			
1	NEWEVENT	MYEVENT	YOUREVENT	HISEVENT
2	DG-B			
3	TANK			
4	C-MOV-1			
5	C-MOV-A	C-MOV-B		
6	C-MOV-A	C-CV-B		
7	C-MOV-A	C-PUMP-B		
8	DG-A	C-MOV-B		
9	DG-A	C-CV-B		
10	DG-A	C-PUMP-B		
11	C-PUMP-A	C-MOV-B		
12	C-PUMP-A	C-CV-B		
13	C-PUMP-A	C-PUMP-B		
14	C-CV-A	C-MOV-B		
15	C-CV-A	C-CV-B		

Figure 57. Addition to cut sets complete.

After adding the new cut set, we decided to move the cut set into location 7. We did this by moving the cursor into the first row and column or over the number 1 and hitting the "m" key, which was understood by the Editor as the Modify command or Change Cut Set. This cleared the field where the number for the cut set was, and we were able to enter in the number 7, as shown in Figure 58. Figure 59 shows that our new cut set is now moved to location 7.

CUT SET EDITOR				
Change Cut Set Type in a new number for the selected Cut Set Use <Enter> when done, and <Esc> to abort				
Set #	Event Names			
7	NEWEVENT	MYEVENT	YOUREVENT	HISEVENT
2	DG-B			
3	TANK			
4	C-MOV-1			
5	C-MOV-A	C-MOV-B		
6	C-MOV-A	C-CV-B		
7	C-MOV-A	C-PUMP-B		
8	DG-A	C-MOV-B		
9	DG-A	C-CV-B		
10	DG-A	C-PUMP-B		
11	C-PUMP-A	C-MOV-B		
12	C-PUMP-A	C-CV-B		
13	C-PUMP-A	C-PUMP-B		
14	C-CV-A	C-MOV-B		
15	C-CV-A	C-CV-B		

Figure 58. Modify cut set position.

CUT SET EDITOR				
Exit / Add / Modify / Delete / Locate / Next / Previous Insert / Replace / Search / Options / Undo Delete / Esc for more help				
Set #	Event Names			
7	NEWEVENT	MYEVENT	YOUREVENT	HISEVENT
8	DG-A	C-MOV-B		
9	DG-A	C-CV-B		
10	DG-A	C-PUMP-B		
11	C-PUMP-A	C-MOV-B		
12	C-PUMP-A	C-CV-B-B		
13	C-PUMP-A	C-PUMP-B		
14	C-CV-A	C-MOV-B		
15	C-CV-A	C-CV-B		
16	C-CV-A	C-PUMP-B		

Figure 59. Cut set number 1 moved to position 7.

To show how to change the name of an event, we positioned the cursor over the event we wished to change and hit the "m" key. This brought up the change Event Name message, and the cursor was placed at the beginning of the field. We then were able to edit the name of the event by using the cursor keys, the Ins key, and the Del key. Figures 60 and 61 show the result of the operation.

C U T S E T E D I T O R

_____Change Event Name_____

Event names which do not exist will be added to the events list
 _____Use <Enter> when done, and <Esc> to abort_____

Set #	Event Names			
1	DG-B			
2	TANK			
3	C-MOV-1			
4	C-MOV-A	C-MOV-B		
5	C-MOV-A	C-CV-B		
6	C-MOV-A	C-PUMP-B		
7	NEWEVENT	YOUREVENT	HISEVENT	
8	DG-A	C-MOV-B		
9	DG-A	C-CV-B		
10	DG-A	C-PUMP-B		
11	C-PUMP-A	C-MOV-B		
12	C-PUMP-A	C-CV-B		
13	C-PUMP-A	C-PUMP-B		
14	C-CV-A	C-MOV-B		
15	C-CV-A	C-CV-B		

Figure 60. Modify event name message.

CUT SET EDITOR				
Exit / Add / Modify / Delete / Locate / Next / Previous Insert / Replace / Search / Options / Undo Delete / Esc for more help				
Set #	Event Names			
1	DG-B			
2	TANK			
3	C-MOV-1			
4	C-MOV-A	C-MOV-B		
5	C-MOV-A	C-CV-B		
6	C-MOV-A	C-PUMP-B		
7	OLDEVENT	YOUREVENT	HISEVENT	
8	DG-A	C-MOV-B		
9	DG-A	C-CV-B		
10	DG-A	C-PUMP-B		
11	C-PUMP-A	C-MOV-B		
12	C-PUMP-A	C-CV-B		
13	C-PUMP-A	C-PUMP-B		
14	C-CV-A	C-MOV-B		
15	C-CV-A	C-CV-B		

Figure 61. Event name NEWEVENT changed.

Next we observed how the three commands Locate, Next, and Previous functioned. By typing the "l" key, the message and prompt for the Locate command appeared, and we then entered the string "??-*". The cursor then moved to the first event name whose third character was the "-", at line 1 column 1 highlighting the name DG-B. (Actually we could have entered in the name of the event which we were looking for. However, the Cut Set Editor will allow for wild card search strings. The "*" matches any characters for any length, and the "?" matches any single character in the position where it was found.) Then we hit the key "n" and noticed that the cursor moved to the next occurrence of our search string. Finally, we typed "p" and the cursor moved back to the previous event name. This indicated that Locate searches for the first occurrence of a given search string, beginning from the top of the file. Also, Next searches for the next occurrence of that same search string. And Previous searches for the previous occurrence of the same search string. (You might think of Previous as a backwards Next.)

Next we skipped down on our line of commands to the Search command, to see how this command differs from Locate. By entering in the "s" key, we were prompted again with a message and asked to enter in a search string. Upon entering the string DG-A we noticed that the cursor move as soon as we had typed the letter "D". This

command is called Speed Search, and it attempts to find a match for your string as soon as it has enough characters to begin a search. After the letter "A" was entered we noticed that the search for this event was found. This function differs from the Locate function in that it begins its search as soon as you start typing in characters. And the search no longer begins from the top of the file but rather from the top of the window or page. Search also updates the Locate search string. For example, lets say you were somewhere in the middle of your cut set data which contained about 400 cut sets. And you were using Next to search for occurrences of a certain event name, when you decided to change the search string. Rather than using Locate to change the string and search back down to where you are positioned, (remember that Locate begins at the beginning of the data,) you can use Search to look for the occurrence of the new search string. The Locate search string will now be updated. Then Next will begin looking for this new search string beginning from where you were before.

Subsequently, we decided to demonstrate the Insert and the Replace commands. We entered the key "i", and the insert messages and prompt appeared just below the title bar. We then entered in the search string and the new event name which was to be added, as shown in Figure 62. We saw then (see Figure 63) that where the occurrence of the search string was found the new event was added to that cut set. So the Insert command allowed us to search globally for an occurrence of a search string and insert into that cut set a new event name.

C U T S E T E D I T O R

Find and Insert

Find *MOV?A
and Insert X

Use <Tab> to move between fields, <Enter> when done and, <Esc> for help

Set #	Event Names			
1	DG-B			
2	TANK			
3	C-MOV-1			
4	C-MOV-A	C-MOV-B		
5	C-MOV-A	C-CV-B		
6	C-MOV-A	C-PUMP-B		
7	OLDEVENT	YOUREVENT	HISEVENT	
8	DG-A	C-MOV-B		
9	DG-A	C-CV-B		
10	DG-A	C-PUMP-B		
11	C-PUMP-A	C-MOV-B		
12	C-PUMP-A	C-CV-B		
13	C-PUMP-A	C-PUMP-B		
14	C-CV-A	C-MOV-B		
15	C-CV-A	C-CV-B		

Figure 62. Insert message and prompt.

CUT SET EDITOR			
Exit / Add / Modify / Delete / Locate / Next / Previous Insert / Replace / Search / Options / Undo Delete / Esc for more help			
Set #	Event Names		
1	DG-B		
2	TANK		
3	C-MOV-1		
4	C-MOV-A	C-MOV-B	X
5	C-MOV-A	C-CV-B	X
6	C-MOV-A	C-PUMP-B	X
7	OLDEVENT	YOUREVENT	HISEVENT
8	DG-A	C-MOV-B	
9	DG-A	C-CV-B	
10	DG-A	C-PUMP-B	
11	C-PUMP-A	C-MOV-B	
12	C-PUMP-A	C-CV-B	
13	C-PUMP-A	C-PUMP-B	
14	C-CV-A	C-MOV-B	
15	C-CV-A	C-CV-B	

Figure 63. The new event X added to three cut sets.

Following that, we selected the "r" and the Replace prompt and message popped up. We then demonstrated how the Replace command searched for an occurrence of a string (wild card strings allowed) and replaced that occurrence with the new string. In our example we showed that you could also use Replace to delete those occurrences when found. Figures 64 and 65 show this.

Note: When both the prompts for Insert and Replace are on the screen, the <Esc> key acts as a help key. When depressed, a help message appears.

C U T S E T E D I T O R

Find X
Find-and-Replace

Replace with

Use <Tab> to move between fields, <Enter> when done and, <Esc> for help

Set #	Event Names		
1	DG-B		
2	TANK		
3	C-MOV-1		
4	C-MOV-A	C-MOV-B	X
5	C-MOV-A	C-CV-B	X
6	C-MOV-A	C-PUMP-B	X
7	OLDEVENT	YOUREVENT	HISEVENT
8	DG-A	C-MOV-B	
9	DG-A	C-CV-B	
10	DG-A	C-PUMP-B	
11	C-PUMP-A	C-MOV-B	
12	C-PUMP-A	C-CV-B	
13	C-PUMP-A	C-PUMP-B	
14	C-CV-A	C-MOV-B	
15	C-CV-A	C-CV-B	

Figure 64. Replace message and prompt.

CUT SET EDITOR				
Exit / Add / Modify / Delete / Locate / Next / Previous Insert / Replace / Search / Options / Undo Delete / Esc for more help				
Set #	Event Names			
1	DG-B			
2	TANK			
3	C-MOV-1			
4	C-MOV-A	C-MOV-B		
5	C-MOV-A	C-CV-B		
6	C-MOV-A	C-PUMP-B		
7	OLDEVENT	YOUREVENT	HISEVENT	
8	DG-A	C-MOV-B		
9	DG-A	C-CV-B		
10	DG-A	C-PUMP-B		
11	C-PUMP-A	C-MOV-B		
12	C-PUMP-A	C-CV-B		
13	C-PUMP-A	C-PUMP-B		
14	C-CV-A	C-MOV-B		
15	C-CV-A	C-CV-B		

Figure 65. Result after the replace operation.

Next we wished to show how the Delete function operated. The Delete command is like the Add and Modify commands in that it is column-dependent. If the cursor is in the number column then the entire cut set is removed from the data. Conversely, if the cursor is in any other column then that event name, if it exists, is removed from its cut set. This we demonstrated by moving the cursor over the event name OLDEVENT and hitting the delete key "d". The event name was then immediately taken from the cut set and the window was adjusted accordingly. Before we went further, we decided to see what the Undo Delete command did and hit the key. We observed that the deletion which just took place was restored. This is the functionality of the Undo Delete. However, it will only work precisely after a deletion has been made. We then moved our cursor into the number column so that it was over the number 7. We hit the delete key again, and this time the cut set was removed from the data. Here we noted that the Undo Delete will also work.

The Delete command also gave us a Veto option just before each operation was completed. We were prompted with the message as shown in Figure 66 and asked to Modify the data, Skip this operation, Continue the operation and not ask again, or End the operation. By default, Veto for deletions is turned on when you enter the Cut Set Editor. There is a Veto for each operation who modifies the cut set data. The default veto for all operations is

off when first entering the Editor, save the Delete command as explained above. If you forget the functionality of Veto, you can hit the <Esc> key and a help message appears whenever the Veto message is on the screen. These Veto defaults may be set using the Options command.

The Options command gives us the ability to set a few important defaults associated with the Cut Set Editor. We illustrated this by typing the "o" key on the keyboard. The Options menu appeared underneath the title bar, and we were then asked to change, if needed, the default settings. The defaults listed were:

- o Range. By default, Range is set to 9999 or a very large maximum number. Range is use by the Insert and Replace commands. These commands begin their operations from the top of the current window and work down "Range" number of cut sets.
- o Veto. Veto tells each operation to ask again if the user wishes to modify the cut set data. By default this is turned off. If you are not familiar with the Cut Set Editor, it may be useful to turn this on when starting the Editor.
- o Veto Delete. Functions the same as Veto above. However, it applies only to the Delete command. Its default is on.
- o Help. Help allows the user to turn off the command and cursor help menus. The operations function as before. However, the accomplished user may wish to have these menus turned off. There is a help message for each field in this menu. By hitting the <Esc> key when in the field, a message associated with that default will appear. Hitting <Esc> again will hide the message. Figure 67 shows what the Options menu looks like.

Finally, when we were done with the editing of the cut sets we hit the Exit command or "e". The screen cleared and we were asked to save or discard the changes we had made (see Figure 68). The user must respond with a "Y" or "N" in order to continue with the next menu. We entered "N" because we left the data the same as they were when we started.

CUT SET EDITOR			
<div style="border: 1px solid black; display: inline-block; padding: 2px 10px;"> Veto Option M Modify / Skip / Continue to end / End Select an option and press <Enter> when done, and <Esc> for help </div>			
Set #	Event Names		
1	DG-B		
2	TANK		
3	C-MOV-1		
4	C-MOV-A	C-MOV-B	
5	C-MOV-A	C-CV-B	
6	C-MOV-A	C-PUMP-B	
7	OLDEVENT	YOUREVENT	HISEVENT
8	DG-A	C-MOV-B	
9	DG-A	C-CV-B	
10	DG-A	C-PUMP-B	
11	C-PUMP-A	C-MOV-B	
12	C-PUMP-A	C-CV-B	
13	C-PUMP-A	C-PUMP-B	
14	C-CV-A	C-MOV-B	
15	C-CV-A	C-CV-B	

Figure 66. Veto Delete menu.

Options	
<div style="border: 1px solid black; display: inline-block; padding: 2px 10px;"> Range value =>9999 Veto <ON> =>N Delete Veto <ON> =>Y Help <ON> =>Y Use <Tab> to move between fields, <Enter> when done, and <Esc> for help </div>	

Figure 67. Options menu.

<div style="border: 1px solid black; padding: 10px; margin: 0 auto; width: 80%;"> <p>Do you want to</p> <p>SAVE the changes</p> <p>made to the Cut Sets</p> <p>(Y/N) </p> </div>

Figure 68. Save Option menu.

3.6 Identifying Accident Sequences

In the previous sections, we discussed how the logic models were created, reduced, and quantified for each of the top events in our simple event tree model of Figure 9. In this section, we explain how we completed our next step, identifying the accident sequences.

We exited through the menus back to the IRRAS 2.0 main menu. We selected the MODIFY Data Base module which brought up the Edit Data Base Records menu (Figure 12). From this menu, we selected option 2 - Event Trees & Sequences by pressing 2, highlighting the DEMO FAMILY, and pressing <Enter>. IRRAS 2.0 then called up the Edit Event Tree menu shown in Figure 69.

Edit Event Tree

Currently Selected Family —> DEMO FAMILY

Option | | Exit / Add / Modify / Delete / Locate / Next / Previous
Sequence Edit

Event Tree Number ----

Event Tree Name

Event Tree Description

Note: The Add options adds a new event tree record. The Modify and Delete options edit the displayed event tree record. The Sequence Edit option allows editing of associated sequences.

Figure 69. Edit Event Tree menu.

With this menu and those to follow, we defined for the IRRAS 2.0 data base the event tree for our example problem shown in Figure 9. We also defined each of the accident sequences in our event tree in terms of failed and successful top events. With this information, IRRAS 2.0 can determine the minimal cut sets for each accident sequence and quantify the event tree.

On the Edit Event Tree menu in Figure 69, we defined an event tree by selecting the Add option. We pressed the letter A, the highlight moved to the Event Tree Name field where we typed in the name of our event tree, LOSP. Event tree names may be up to 16 characters long. Next, we used the <Down Arrow> key to move the highlight to the description field. We typed in a description of our event tree not exceeding 60 characters. When we were satisfied with the form, we pressed <Enter> to send the information to the data base. IRRAS 2.0 acknowledged this addition to the data base by displaying the message

Record Added

at the bottom of the screen and moving the cursor to the option field. Our screen looked like Figure 70.

E d i t E v e n t T r e e

Currently Selected Family —> DEMO FAMILY

Option |A| Exit / Add / Modify / Delete / Locate / Next / Previous
Sequence Edit

Event Tree Number 1

Event Tree Name LOSP

Event Tree Description
Loss of offsite power event tree.

Note: The Add options adds a new event tree record. The Modify and Delete options edit the displayed event tree record. The Sequence Edit option allows editing of associated sequences.

Figure 70. LOSP Event Tree data.

The next step was to define the accident sequences associated with this event tree. We did this by selecting S for Sequence Edit and pressing <Enter>. IRRAS 2.0 displayed the Edit Sequence menu with the Family and Event Tree fields already filled in, as shown in Figure 71.

Edit Sequence				
Option Exit / Add / Modify / Delete / Locate / Next / Previous				
Include Fault Trees in Sequence				
Family DEMO FAMILY	Event Tree	LOSP	Sequence	
Description				
5th % -----E----	Median -----E----	Min -----E----	Std. Dev -----E----	
95th % -----E----	Mean -----E----	Max -----E----	Skewness -----E----	
Min Cut Upper Bound -----E----			Kurtosis -----E----	

Figure 71. Edit Sequence menu.

From the event tree diagram of Figure 9, we noted that our model contained three sequences. These sequences were

- o Sequence 1. A loss of offsite power, successful operation of the Emergency Cooling System, the core remained cooled, no core damage, no challenge to the Containment Cooling System.
- o Sequence 2. A loss of offsite power, failure of the Emergency Cooling System caused core damage, success of the Containment Cooling System ensured only a small release.
- o Sequence 3. A loss of offsite power, failure of the Emergency Cooling System caused core damage, failure of the Containment Cooling System allowed a large release.

To add these sequences to the LOSP event tree, we chose the Add option by pressing the letter A. The highlight moved to the Sequence field, where we typed in the name of the first sequence. Sequence names can be up to 16 characters. Then we used the <Down Arrow> key to move the highlight to the Description field. Here we typed in a sequence description not exceeding 60 characters. The remaining fields were left blank for IRRAS 2.0 to fill in after the sequence was analyzed. When we were satisfied with the name and description, we pressed the <Enter> key to add the sequence to the data base. IRRAS 2.0 replied with the

Record Added

message and moved the cursor to the option field. At this point our screen looked like Figure 72.

Edit Sequence

Option | | Exit / Add / Modify / Delete / Locate / Next / Previous
Include Fault Trees in Sequence

Family DEMO FAMILY Event Tree LOSP Sequence 1-LOSP-ECS
Description Loss of Offsite Power with success of ECS, no core damage.

5th % -----E----	Median -----E----	Min -----E----	Std. Dev -----E----
95th % -----E----	Mean -----E----	Max -----E----	Skewness -----E----
Min Cut Upper Bound -----E----		Kurtosis -----E----	

Figure 72. Sequence 1 added to the data base.

With the sequence added to the data base, we were ready to define the logic of the sequence in terms of failed and successful top events. To do this, we selected Include Fault Trees in Sequence, Option I. This gave us the Include Fault Trees menu as shown in Figure 73.

Include Fault Trees

Option |I| Exit / Include Fault Tree in Sequence

Family DEMO FAMILY Event Tree LOSP Sequence 1-LOSP-ECS
Description Loss of Offsite Power with success of ECS, no core damage.

	FAULT	TREES	
CCS	ECS	LOSP	

Note: To include a fault tree () or its compliment () or remove it from the sequence, use <I> option. Use <PgUp> or <PgDn> to display more trees.

Figure 73. Include Fault Trees menu for Sequence 1-LOSP-ECS.

The top 4 fields were already filled in by IRRAS 2.0. The lower section of this menu displayed all the zero-level fault trees in the DEMO FAMILY. Option I, Include Fault Tree in Sequence, was already selected, so we used the <Down Arrow> key to move the highlight to the fault tree list. This list was used to indicate which fault trees were included in the sequence and whether the system was successful or failed. A successful system is indicated by an up arrow () in front of the fault tree name. A failed system is indicated by a down arrow (). A blank space in front of the fault tree name means that the fault tree is not included in the sequence.

When the highlight was moved to the fault tree list, the first fault tree highlighted was the ECS tree. In Sequence 1, the ECS system is successful. To place an up arrow in front of the ECS tree, we simply hit the <Enter> key two times. The first time put a down arrow in front of ECS, the second time changed the down arrow to an up arrow. With the <Right Arrow> key, we moved the highlight to the LOSP tree. We hit the <Enter> key once to include the initiating event as a failed system in the sequence. Since Sequence 1 does not include the CCS fault tree (no branch point at the CCS top event), the space in front of the CCS tree was left blank. Our screen then looked like Figure 74.

Include Fault Trees

Option |I| Exit / Include Fault Tree in Sequence

Family DEMO FAMILY Event Tree LOSP Sequence 1-LOSP-ECS
 Description Loss of Offsite Power with success of ECS, no core damage.

CCS	ECS	LOSP	

Note: To include a fault tree () or its compliment () or remove it from the sequence, use <I> option. Use <PgUp> or <PgDn> to display more trees.

Figure 74. Logic for Sequence 1.

We used the <Up Arrow> key to move the highlight to the Option field and selected E to Exit this menu. Exiting caused the logic for the sequence to be added to the data base and caused IRRAS 2.0 to return to the Edit Sequence menu. From here, we were able to repeat the processes of sequence identification and logic assignment for Sequences 2 and 3. When this was completed, we exited through the menus back to the IRRAS 2.0 main menu.

3.7 Sequence Analysis

The accident sequences were ready to process at the end of Section 3.6. The Sequence Analysis module was accessed from the IRRAS 2.0 main menu by selecting the SEQUENCE Analysis option. This brought up the IRRAS-PC Sequence Analysis System menu. This menu is shown in Figure 75.

Figure 75. IRRAS-PC Sequence Analysis System menu.

3.7.1 Sequence Cut Set Generation and Quantification

We did not use the Modify Event Data option because all the event data were set properly during the fault tree analysis and no changes were needed. Therefore, we went directly into the Analyze Sequences option. IRRAS 2.0 put the Analyze Sequences menu of Figure 76 on the screen.

Figure 76. Analyze Sequences menu.

This menu displayed the names of the sequences defined earlier, along with their descriptions, and, to the far left of the display, a set of flags indicating the status of the analysis of each sequence. The meanings of the flags are

- e - The event tree graphics do not match the logic of the sequences presently in the data base. Since IRRAS 2.0 does not yet allow event tree graphics, this flag will always be present.
- c - The last generated cut sets for the sequence do not match the sequence logic presently in the data base.
- q - The last quantification does not match the sequence cut sets or basic event data presently in the data base.
- u - The last uncertainty analysis does not match the cut sets or basic event data presently in the data base.

To begin the analysis, we chose option C - Cut Set Generation by pressing the letter C, highlighting the first sequence, and

pressing <Enter>. This brought up the Cut Set Generation Cutoff Values screen shown in Figure 77, and put the

Change to desired cutoff values and press <Enter>
message at the bottom of the screen.

A n a l y z e S e q u e n c e s

Currently Selected Family —> DEMO FAMILY

Option |C| Exit / Cut Set Generation / Quantification / Uncertainty calc.

CUT SET GENERATION CUTOFF VALUES

Perform Probability Cutoff? (Y/N) Y	Cutoff Value	1.000E-015
Perform Cut Set Size Cutoff? (Y/N) Y	Size Cutoff	6
Mission Time	2.400E+001	

Figure 77. Sequence Cut Set Generation Cutoff Values menu.

This menu was used to select our cutoff criteria for the generation of the sequence minimal cut sets. Cutoff criteria were used to limit the number of cut sets generated to a manageable number and to speed up the processing time. For extremely large or complex sequences, cutoff criteria may be required to allow IRRAS 2.0 to process the sequence.

The probability cutoff tells IRRAS 2.0 to not generate any cut sets with a probability less than the specified value. The size cutoff tells IRRAS 2.0 to not generate any cut sets with more basic events than the specified value. The defaults for these criteria are probability cutoff in effect with a minimum cut set probability of 1.0E-15 and size cutoff in effect with a maximum cut set size of 6 basic events per cut set. These defaults can be permanently changed in the UTILITY Options module (Define Constants). Changes for just the selected tree can be made on this menu. We opted to not truncate on the number of basic events per cut set by using the <Down Arrow> key to move the highlight to the Perform Cut Set Size Cutoff (Y/N) field and pressing <N> and <Enter>.

IRRAS 2.0 began processing the sequence and kept us informed of the status of the analysis at each step. It indicated which sequence was being processed, the number of intermediate cut sets that had been generated, when it was supersetting the cut sets, when it was sorting the cut sets, and when it was quantifying them. When all this was completed, IRRAS 2.0 provided a summary of the cut set generation and quantification. For our analysis of sequence 1-LOSP-ECS, the summary looked like Figure 78.

Analyze Sequences		
Currently Selected Family —> DEMO FAMILY		
Option C Exit / Cut Set Generation / Quantification / Uncertainty calc.		
Results		
Sequence Name	1-LOSP-ECS	
Cut Set Size		
1	1	
2	0	Min Cut Upper Bound
3	0	2.300E+000
4	0	
5	0	Elapsed Time
6	0	00:00:01.040
7	0	
8	0	
9	0	
10	0	
>10	0	
Total	1	

Figure 78. Cut set generation summary for Sequence 1-LOSP-ECS.

The summary showed that sequence 1-LOSP-ECS generated only one single failure cut set. The point estimate calculation of the minimal cut set upper bound was 2.30 and the total time required for the processing was 1.04 seconds. More details of the analysis are available in the Display Sequences option in the SEQUENCE Analysis module and in the REPORT on Data Base module.

IRRAS provided a message stating

Press <Enter> key to continue

when it put up the summary of the analysis. So when we were finished with the summary, we pressed <Enter>. IRRAS 2.0 indicated that the cut set generation was completed and returned us to the Analyze Sequences menu. The flags next to sequence 1-LOSP-ECS were

updated to show only the uncertainty analysis remaining to be completed. This process was repeated for each accident sequence. When all sequences had their cut sets generated, the Analyze Sequences menu looked like Figure 79.

Analyze Sequences

Currently Selected Family —> DEMO FAMILY

Option |C| Exit / Cut Set Generation / Quantification / Uncertainty calc.

SEQUENCES

e	u-->	1-LOSP-ECS	Loss of Offsite Power with success of ECS, no core da
e	u-->	2-LOSP-ECS-CCS	Loss of Offsite Power, failed ECS, successful CCS, sm
e	u-->	3-LOSP-ECS-CCS	Loss of Offsite Power, failed ECS, failed CCS, lg. re

Note : Changed indicators precede each sequence with meanings as follows-
 e-Event Trees, c-Cut Sets, q-Quantification, u-Uncertainty
 More change information may be obtained for each seq. using <Esc>.
 Use <PgUp> and <PgDn> to display more sequences.

Figure 79. Analyze Sequences menu after cut set generation.

NOTE: Selecting a sequence analysis option without highlighting a specific sequence will cause IRRAS 2.0 to process all sequences needing the specified type of analysis. IRRAS 2.0 will then provide a warning and allow the user to verify his intent. If the batch operation was inadvertently selected, the user can return to the previous menu by indicating no desire to continue.

The cut sets for the remaining sequences were also generated as just explained.

3.7.2 Performing Sequence Uncertainty Analyses

After generating the minimal cut sets, we were ready to perform the uncertainty analyses. We started by selecting U for the Uncertainty calculation option on the Analyze Sequences menu, highlighting the 1-LOSP-ESC sequence and pressing <Enter>. This brought up the Uncertainty Calculation Values menu shown in Figure 80.

A n a l y z e S e q u e n c e s

Currently Selected Family —> DEMO FAMILY

Option |U| Exit / Cut Set Generation / Quantification / Uncertainty calc.

—UNCERTAINTY CALCULATION VALUES—

Number of samples to use in Monte Carlo simulation	1000
Seed for random number generator	3571
Note: Use "0" as the seed to get a random seed from the clock	

Change to desired uncertainty values and press Enter

Figure 80. Sequence Uncertainty Calculation Values menu.

On this menu, we set the desired number of samples for the uncertainty analysis and the seed value for the random number generator. Figure 80 shows the default values set into IRRAS 2.0. These defaults can be permanently changed in the UTILITY Options module (Define Constants) or changed for just this sequence on this menu. Since we were satisfied with the defaults, we pressed <Enter>. IRRAS 2.0 flashed the message

Reading in basic events - please stand by

IRRAS 2.0 began the Monte Carlo analysis and provided the status along the way. The screen showed the name of the sequence being processed, the total number of samples chosen, and the current sample being calculated. For each sample, IRRAS 2.0 shows the value calculated for the mean value of the distribution. Upon completion of the uncertainty analysis, IRRAS 2.0 displayed a summary of the results. The summary for the 1-LOSP-ECS analysis is shown in Figure 81.

Figure 81. Uncertainty results summary for Sequence 1-LOSP-ECS.

The summary provided the boilerplate information such as the sequence name, the number of minimal cut sets for the sequence, the number of basic events in those cut sets, the random seed used, and the number of samples calculated. The summary also gave the results of the calculations including the point estimate calculation; the mean of the distribution for the sequence; the 5th percentile, median, and 95th percentile values of the distribution; the minimum and maximum sample values; and the various statistical moments of the distribution. For a more detailed discussion of these parameters, see Section 6.4.

When we finished reviewing the summary results, we pressed <Enter>. IRRAS 2.0 indicated that the uncertainty analysis was completed and the Analyze Sequences menu returned. We performed the uncertainty analyses for the remaining sequences using the method just described.

The Uncertainty analysis option has the ability to perform batch analyses on all the sequences that are not up-to-date. This works just like the batch mode for cut set generation. Select the uncertainty analysis without highlighting a fault tree and answer yes <Y> to the desire to continue question. IRRAS 2.0 will then process one sequence after another using the specified random seed and number of samples for each sequence.

3.7.3 Displaying the Sequence Analysis Results

At this point, we had completed the creation of the accident sequence models, generated the minimal cut sets, quantified the models and performed uncertainty analyses. We next desired to look at our results. (Actually we looked at our results at every step of the analysis, which seems to be a good practice, but it is more convenient to discuss how we did it now.)

To look at our results, we exited from the Analyze Sequences menu back to the IRRAS-PC Sequence Analysis System menu. From this menu, we chose the Display Sequences option by pressing D <Enter>. This brought up the Display Sequences menu shown in Figure 82

Display Sequence Data

Currently Selected Family —> DEMO FAMILY

Option |E| Exit / Cut Sets / Importance measures / Uncertainty data

SEQUENCES

1-LOSP-ECS	Loss of Offsite Power with success of ECS, no core damage.
2-LOSP-ECS-CCS	Loss of Offsite Power, failed ECS, successful CCS, sm. rel.
3-LOSP-ECS-CCS	Loss of Offsite Power, failed ECS, failed CCS, lg. release.

Note : Use <PgUp> and <PgDn> to display more sequences.

Figure 82. Display Sequence Data menu.

From this menu, we were able to look at the analyses results in more detail. Options on this menu provided us with the ability to look at the sequence cut sets and quantification, the uncertainty analysis results, and some importance measures of the basic events.

To review the sequence cut sets, we chose the Cut sets option by pressing C, highlighting the desired sequence, and pressing <Enter>. IRRAS 2.0 displayed

Loading event data

and then brought up the Sequence Cut Sets display shown in Figure 83.

SEQUENCE CUT SETS						
Family Name DEMO FAMILY			Sequence Name 2-LOSP-ECS-CCS			
Min Cut	2.300E+000	Num 1	Part.==>	2.300E+000	100.00%	Num 1
Option [B] Exit/ Partition/ Report/ Basic Events <= choose a cut set below						
Num	%	Frequency	Event Names			
1	95.16	4.600E-002	DG-A	IE-LOSP		
2	4.76	2.300E-003	E-MOV-1	IE-LOSP		
3	0.12	5.750E-005	E-MOV-B	E-MOV-A	IE-LOSP	
4	0.07	3.450E-005	E-PUMP-B	E-MOV-A	IE-LOSP	
5	0.07	3.450E-005	E-MOV-B	E-PUMP-A	IE-LOSP	
6	0.04	2.070E-005	E-PUMP-B	E-PUMP-A	IE-LOSP	
7	0.00	1.150E-006	E-CV-B	E-MOV-A	IE-LOSP	
8	0.00	1.150E-006	E-MOV-B	E-CV-A	IE-LOSP	
9	0.00	6.900E-007	E-PUMP-B	E-CV-A	IE-LOSP	
10	0.00	6.900E-007	E-CV-B	E-PUMP-A	IE-LOSP	
11	0.00	2.300E-008	E-CV-B	E-CV-A	IE-LOSP	
Use <PgUp> or <PgDn> to display more Cut Sets						

Figure 83. Sequence Cut Sets display.

The top of this display gave the family and sequence names along with the minimal cut set upper bound value as calculated by the point estimate quantification. The lower portion showed the quantified minimal cut sets. The first column gave the ranking of the cut set according to probability. The second column indicated an approximate percentage contribution of the cut set to the minimal cut set upper bound. The third column gave the frequency of each individual cut set and the last column gave a listing of the basic events making up each cut set. More cut sets were reviewed by using the <PgUp> and <PgDn> keys.

We could have performed some additional analysis on the cut sets by selecting the Partition option. This option allows us to select the cut sets to be displayed by limiting them to only those cut sets containing selected or qualified events. For a more detailed description of this option, refer to section 3.5.3.

More information on specific cut sets was available by choosing the Basic Event option, highlighting one of the cut sets,

and pressing <Enter>. This called up the Cut Set Basic Event Descriptions display. Figure 84 shows this display.

CUT SET BASIC EVENT DESCRIPTIONS						
Family DEMO FAMILY		Sequence 2-LOSP-ECS-CCS			Min Cut 4.834E-002	

Event Name	Description					
	Probability	Comp ID	System	Type	FM	Location
DG-A	Emergency diesel generator A					
	2.000E-002	EP-DG-A	EP	DG	C1	FZ6
IE-LOSP	Loss of Offsite Power Initiation Event					
	2.300E+000				IE	

Use <PgUp> or <PgDn> to display more Events
Press <Enter> to Return

Figure 84. Sequence Cut Set Basic Event Descriptions display.

This display provided all the available information on each basic event contained in the chosen cut set. In our example, the only additional information was the basic event probability. Had we given the basic events descriptions and classified them according to system, component, failure mode and location, this information would have also been displayed. The assignment and use of these types of basic event attributes is discussed in Section 4.2.1.1.

We returned to the Display Sequence Data menu by pressing <Enter>, selecting E for Exit and pressing <Enter> again. From here we were able to look at the uncertainty results for any and all sequences in the family by selecting the Uncertainty data option, U, highlighting the desired sequence, and pressing <Enter>. This brought up the Uncertainty Data display shown in Figure 85.

UNCERTAINTY DATA

Family DEMO FAMILY
 Sequence 2-LOSP-ECS-CCS
 Description Loss of Offsite Power, failed ECS, successful CCS, sm. rel.

Mean	5.244E-002	5th %	4.091E-004	Std. Dev	1.530E-001
Mincut	4.834E-002	Median	8.104E-003	Minimum	1.099E-005
		95th %	1.838E-001	Maximum	6.741E+000

Press <Enter> to return to previous form.

Figure 85. Sequence Uncertainty Data display.

This display gave us the most important results from the uncertainty analysis. Pressing <Enter> returned us to the Display Sequence Data menu.

The last display of the analysis results we used was the basic events importance measures. This option was selected by pressing I, highlighting the desired sequence, and pressing <Enter>. IRRAS 2.0 flashed several messages on the screen that reported the status of loading the cut sets, calculating the importance measures, and sorting the basic events. Then IRRAS 2.0 put the Basic Event Importance display on the screen. The screen looked like Figure 86.

Figure 86. Sequence Basic Event Importance display.

This display showed every basic event in the chosen sequence. For each basic event, IRRAS 2.0 calculated and displayed three importance measures. These importance measures are

- o Fussell-Vesely Importance, an indication of the percentage of the minimal cut set upper bound contributed by the basic event.
- o Risk Reduction Ratio, an indication of how much the minimal cut set upper bound would decrease if the basic event were made perfect (never fails).
- o Risk Increase Ratio, an indication of how much the minimal cut set upper bound would go up if the basic event were always failed.

The Basic Event Importance display listed the basic event name, its failure probability, and the three importance measures. The basic events were listed in order of Fussell-Vesely importance, high to low. For a more detailed discussion of importance measures, see Section 5.5.

3.8 Getting Reports on the Analysis

In addition to displays of the analysis, IRRAS 2.0 can generate reports on any and all portions of the data base. These reports can be reviewed on the screen, sent directly to a printer, or placed in a file on the hard disk for later use.

NOTE: Before generating a report, the user should initialize the printer by moving the paper to the top of form position, then turning the printer off and then back on to set the top of form.

The available reports are generated in the REPORT on Data Base module, accessed from the IRRAS 2.0 main menu.

Selecting the REPORT on Data Base option, we created data base reports to include in the documentation of our analysis. This option brought up the IRRAS Reports menu as shown in Figure 87.

IRRAS REPORTS

Option |E| Exit / Basic Event / Fault Tree / Sequence <= choose a family
 Report of Families <= other options

DEMO FAMILY Demonstration sample family

Note: Select desired report, press <Enter> to continue.
 Use <PgUp> or <PgDn> to see more Family names.

Figure 87. The IRRAS Reports menu.

This menu allowed us to choose reports on the basic events, fault trees, sequences, and families contained in the data base.

3.8.1 Basic Event Reports

To access the reports on the basic events, we selected option B - Basic Event, highlighted the DEMO FAMILY, and pressed <Enter>. IRRAS 2.0 put the Basic Event Reports menu on the screen. This menu is shown in Figure 88.

Figure 88. IRRAS 2.0 Basic Event Reports menu.

3.8.1.1 Basic Events Summary Report This menu lists the three basic event reports available in IRRAS 2.0. The first of these is the summary report. The summary report gives the primary and secondary names of the basic events in the selected family, the component and system designations, the component identification number, the component location code, and the component failure mode designation. An example of the basic events summary report is provided in Figure 89.

BASIC EVENTS SUMMARY REPORT							
Family: DEMO FAMILY							
Event Number	Primary Name	Secondary Name	Component Type	ID	System	Loca- tion	Fail Mode
1	C-CV-A CCS Train A pump discharge check valve	C-CV-A	CV	C-CV-A	CCS	FZ1	A1
2	C-CV-B CCS Train B pump discharge check valve	C-CV-B	CV	C-CV-B	CCS	FZ2	A1
3	C-MOV-1 CCS suction isolation valve	C-MOV-1	MOV	C-MOV-1	CCS	FZ3	A2
4	C-MOV-A CCS Train A pump discharge isolation valve	C-MOV-A	MOV	C-MOV-A	CCS	FZ1	A2
5	C-MOV-B CCS Train B pump discharge isolation valve	C-MOV-B	MOV	C-MOV-B	CCS	FZ2	A2
6	C-PUMP-A CCS Train A motor-driven pump	C-PUMP-A	MDP	C-PMP-A	CCS	FZ1	B1
7	C-PUMP-B CCS Train B motor-driven pump	C-PUMP-B	MDP	C-PMP-B	CCS	FZ2	B1
8	DG-A Emergency diesel generator A	DG-A	DG	EP-DG-A	EP	FZ6	C1
9	DG-B Emergency diesel generator B	DG-B	DG	EP-DG-B	EP	FZ7	C1
10	E-CV-A ECS Train A pump discharge check valve	E-CV-A	CV	E-CV-A	ECS	FZ5	A1
11	E-CV-B ECS Train B pump discharge check valve	E-CV-B	CV	E-CV-A	ECS	FZ4	A1
12	E-MOV-1 ECS suction isolation valve	E-MOV-1	MOV	E-MOV-1	ECS	FZ3	A2
13	E-MOV-A ECS Train A pump discharge isolation valve	E-MOV-A	MOV	E-MOV-A	ECS	FZ5	A2
14	E-MOV-B ECS Train B pump discharge isolation valve	E-MOV-B	MOV	E-MOV-B	ECS	FZ4	A2
15	E-PUMP-A ECS Train A motor-driven pump	E-PUMP-A	MDP	E-PMP-A	ECS	FZ5	B1
16	E-PUMP-B ECS Train B motor-driven pump	E-PUMP-B	MDP	E-PMP-B	ECS	FZ4	B1
17	IE-LOSP Loss of Offsite Power Initiating Event	IE-LOSP	IE				
18	TANK RWST supply to the injection and cooling systems	TANK	TNK	E-TK-1	ECS	FZ3	D1
1987/12/15		16:16:20		page		1	

Figure 89. Basic Events Summary Report.

To get this report, we selected S for Summary from the Basic Event Reports menu. This brought up the IRRAS Reports Information Request form displayed in Figure 90.

I R R A S R E P O R T S
Information Request

Enter the desired title of the report and the name
of the file to which the report will be output.

Title: BASIC EVENTS SUMMARY REPORT

File Name: CON

NOTE: File Name = CON Sends report to the screen.
 PRN Sends report to the printer.
 filename Sends report to the hard disk file.
 blank EXIT

Figure 90. IRRAS Reports Information Request form.

With this form, we entered a title for our report. Figure 74 shows the default title. This can be changed by moving the cursor to the Title field and typing in the new title. This title will be at the top of each page of the report.

IRRAS 2.0 reports can be sent to several different places, depending on the name placed in the File Name field. The default file name is CON. This sends the report to the computer screen. This is the default because it doesn't waste paper or hard disk space if it is inadvertently left unchanged. Changing the file name to PRN sends the report directly to a printer. Making the file name field blank exits the Information Request form and returns the system to the Basic Event Report menu. Entering any other file name that is legal in DOS sends the report to the hard disk in a file with the chosen file name. Figure 91 shows what the Information Request form looked like after we selected a title and designated that the report go to our printer.

I R R A S R E P O R T S
Information Request

Enter the desired title of the report and the name
of the file to which the report will be output.

Title: BASIC EVENTS SUMMARY REPORT

File Name: PRN

NOTE: File Name = CON Sends report to the screen.
 PRN Sends report to the printer.
 filename Sends report to the hard disk file.
 blank EXIT

Figure 91. IRRAS Reports Information Request form completed.

Pressing the <Enter> key caused IRRAS 2.0 to display

Basic Event Report being generated

at the bottom of the screen, followed by

Report Completed

Then the printer started printing out the report and the screen returned to the Basic Event Reports menu (Figure 88). The printed report has already been shown in Figure 89.

3.8.1.2 Basic Events Probabilities Report. The Basic Events Probabilities Report is obtained with the Probabilities option on the Basic Event Reports menu of Figure 88. A title and file name are supplied and the report is generated just as described for the Basic Events Summary Report. The Basic Events Probabilities Report is shown in Figure 92.

BASIC EVENTS PROBABILITY REPORT					
Family: DEMO FAMILY					
Event Number	Primary Name	Calculation Type	Mean Probability	Lambda	Tau
1	C-CV-A	1	1.000E-004	+0.000E+000	+0.000E+000
2	C-CV-B	1	1.000E-004	+0.000E+000	+0.000E+000
3	C-MOV-1	1	1.000E-003	+0.000E+000	+0.000E+000
4	C-MOV-A	1	5.000E-003	+0.000E+000	+0.000E+000
5	C-MOV-B	1	5.000E-003	+0.000E+000	+0.000E+000
6	C-PUMP-A	1	3.000E-003	+0.000E+000	+0.000E+000
7	C-PUMP-B	1	3.000E-003	+0.000E+000	+0.000E+000
8	DG-A	1	2.000E-002	+0.000E+000	+0.000E+000
9	DG-B	1	2.000E-002	+0.000E+000	+0.000E+000
10	E-CV-A	1	1.000E-004	+0.000E+000	+0.000E+000
11	E-CV-B	1	1.000E-004	+0.000E+000	+0.000E+000
12	E-MOV-1	1	1.000E-003	+0.000E+000	+0.000E+000
13	E-MOV-A	1	5.000E-003	+0.000E+000	+0.000E+000
14	E-MOV-B	1	5.000E-003	+0.000E+000	+0.000E+000
15	E-PUMP-A	1	3.000E-003	+0.000E+000	+0.000E+000
16	E-PUMP-B	1	3.000E-003	+0.000E+000	+0.000E+000
17	IE-LOSP	1	2.300E+000	+0.000E+000	+0.000E+000
18	TANK	1	1.000E-007	+0.000E+000	+0.000E+000

1987/12/15 16:17:52 page 1

Figure 92. Basic Events Probabilities Report.

This report contains the following:

- o Header. The title of the report and the family. This is repeated at the top of each page of the report.
- o Event Number. The order in which the events were placed in the data base.
- o Primary Name. The basic event name used in the fault tree. (See Appendix B for more on primary and alternate names.)
- o Calculation Type. The designator for the calculation type selected for the basic event (calculation types 1-9 discussed in Section 3.4.1).

- o Lambda and Tau. The values for the variables in the calculation type equations. These are a failure rate, and a time. (See Section 3.4.1.)
- o Mean Probability. The calculated mean failure probability derived from the calculation type and the input variables.
- o Footer. The footer, at the bottom of each page, contains the date and time that the report was generated and the page number.

3.8.1.3 Basic Events Uncertainty Values Report. The Basic Events Uncertainty Values Report is obtained by selecting the U - Uncertainty Values option on the Basic Event Reports menu. Generating the report is done using the method described for the other basic event reports. Figure 93 is the resulting report.

BASIC EVENTS UNCERTAINTY VALUES REPORT					
Family: DEMO FAMILY					
Event Number	Primary Name	Distribution Type	Mean Probability	Uncertainty Value	Correlation Class
1	C-CV-A	L	1.000E-004	3.000E+000	1
2	C-CV-B	L	1.000E-004	3.000E+000	1
3	C-MOV-1	L	1.000E-003	5.000E+000	3
4	C-MOV-A	L	5.000E-003	5.000E+000	2
5	C-MOV-B	L	5.000E-003	5.000E+000	2
6	C-PUMP-A	L	3.000E-003	5.000E+000	4
7	C-PUMP-B	L	3.000E-003	5.000E+000	4
8	DG-A	L	2.000E-002	1.000E+001	6
9	DG-B	L	2.000E-002	1.000E+001	6
10	E-CV-A	L	1.000E-004	3.000E+000	1
11	E-CV-B	L	1.000E-004	3.000E+000	1
12	E-MOV-1	L	1.000E-003	5.000E+000	3
13	E-MOV-A	L	5.000E-003	5.000E+000	2
14	E-MOV-B	L	5.000E-003	5.000E+000	2
15	E-PUMP-A	L	3.000E-003	5.000E+000	4
16	E-PUMP-B	L	3.000E-003	5.000E+000	4
17	IE-LOSP	L	2.300E+000	1.000E+001	7
18	TANK	L	1.000E-007	3.000E+000	5
1987/12/15		16:19:26		page	1

Figure 93. Basic Events Uncertainty Values Report.

This report contains the following:

- o Header. The title of the report and the family. This is repeated at the top of each page of the report.
- o Event Number. The order in which the events were placed in the data base.
- o Primary Name. The basic event name used in the fault tree. (See Appendix B for more on primary and alternate names.)
- o Distribution Type. The designator for the distribution type selected for the basic event. [Lognormal (L) distribution is the only type allowed at this time.]
- o Mean Probability. The calculated mean failure probability derived from the calculation type and the input variables.
- o Uncertainty Value. For the lognormal distribution, this is the error factor. See Section 3.4.2 for more information.
- o Correlation Class. The correlation class assigned to the basic event. See Section 3.4.2 for more on correlation class designators.
- o Footer. The footer at the bottom of each page contains the date and time that the report was generated and the page number.

3.8.2 Fault Tree Reports

To access the reports on the fault trees, we selected option F - Fault Tree, on the IRRAS Reports menu (Figure 87), highlighted the DEMO FAMILY, and pressed <Enter>. IRRAS 2.0 put the Fault Tree Reports menu on the screen. This menu is shown in Figure 94

IRRAS REPORTS
 Fault Tree Reports

Currently Selected Family => DEMO FAMILY

Option |E| Exit / Logic / eXpanded Logic / Modified Logic /
 Importance Measures / Cut Set / Summary / Uncertainty Values
 All Levels / Zero Levels

	-FAULT-	-TREES-
CCS	Containment Cooling System fault tree	
ECS	Emergency Cooling System fault tree	
LOSP	Loss of Offsite Power Initiating Event fault tree	

Note : Use <PgUp> and <PgDn> to display more fault trees. To execute options (L,X,M,I,C) you must highlight a fault tree.

Figure 94. Fault Tree Reports menu.

This menu lists the fault tree reports available in IRRAS 2.0. These reports are described in the sections that follow.

3.8.2.1 Fault Tree Logic Report. The Fault Tree Logic Report displays the alphanumeric logic file that IRRAS 2.0 automatically created from the graphics input. This is the fault tree logic used as input to the FAULT Tree Analysis module.

The report was obtained by selecting option L - Logic from the Fault Tree Reports menu, highlighting the desired tree, and pressing <Enter>. The IRRAS Reports Information Request form appeared on the screen. This form is shown in Figure 95

IRRAS REPORTS
Information Request

Enter the desired title of the report and the name
of the file to which the report will be output.

Title: FAULT TREE LOGIC REPORT

File Name: CON

NOTE: File Name = CON Sends report to the screen.
 PRN Sends report to the printer.
 filename Sends report to the hard disk file.
 blank EXIT

Figure 95. IRRAS Reports Information Request form
for a Fault Tree Logic Report.

We used this form to determine the title of our report and to direct the output to our printer, just as we did for the basic events reports in Section 3.8.1.1. Figure 96 is an example of the Fault Tree Logic Report.

FAULT TREE LOGIC REPORT		
Family: DEMO FAMILY		
Fault Tree: CCS		
Gate Name	Type	Inputs
C-MOV-1-FAILS	OR	C-MOV-1, DG-B
CCS-SUPPLY	OR	C-MOV-1-FAILS, TANK
CCS-TOP	OR	CCS-SUPPLY, CCS-TRAINS
CCS-TRAIN-A	OR	C-CV-A, C-MOV-A, DG-A, C-PUMP-A
CCS-TRAIN-B	OR	C-CV-B, C-MOV-B, C-PUMP-B, DG-B
CCS-TRAINS	AND	CCS-TRAIN-A, CCS-TRAIN-B
<div> 1987/12/15 16:23:04 page 1 </div>		

Figure 96. Fault Tree Logic Report.

This report contains:

- o Header. The header consists of the report title, the family name, and the fault tree name. The header is repeated at the top of each page of the report.
- o Gate Name. The name assigned to the logic gate.
- o Type. The type of logic gate (OR, AND, TRAN, etc.).
- o Inputs. The names of the basic events or gates under the logic gate.
- o Footer. The footer at the bottom of each page contains the date and time that the report was generated and the page number.

Thus, in our example report, gate CCS-TOP is an OR gate with two inputs under it, CCS-SUPPLY and CCS-TRAINS. Both of these inputs are gates which are defined on the next two lines of the report. Inputs that are not defined somewhere in the logic report are basic events.

3.8.2.2 Fault Tree Expanded Logic Report. The expanded logic report is the same as the logic report except that if any transfer gates exist in the fault tree they are expanded. When the transfer gate is expanded, the logic associated with the transfer is included in the logic report and any transfers in this tree are also expanded. This report is useful if the user gets unresolved transfer errors while trying to process the fault tree. The unresolved transfers will appear as transfer gates in this report.

3.8.2.3 Fault Tree Modified Logic Report. The second fault tree report listed on the Fault Tree Reports menu (Figure 94) is the Fault Tree Modified Logic Report. This report shows the fault tree logic actually processed by IRRAS 2.0 to generate the minimal cut sets. The modified logic differs from the logic created from the graphics in that IRRAS 2.0 coalesces gates and separates out independent subtrees to improve the efficiency of generating the minimal cut sets.

This report is obtained by selecting option M - Modified logic from the Fault Tree Reports menu. A title and directions for the output are supplied as with the other reports. Figure 97 shows this report for the CCS fault tree.

FAULT TREE MODIFIED LOGIC REPORT		
Family: DEMO FAMILY		
Fault Tree: CCS		
Gate Name	Type	Inputs
CCS-TOP	OR	CCS-TRAINS, DG-B, dummyGate1
CCS-TRAINS	AND	CCS-TRAIN-A, CCS-TRAIN-B
dummyGate1	OR	C-MOV-1, TANK
CCS-TRAIN-A	OR	C-CV-A, C-MOV-A, DG-A, C-PUMP-A
CCS-TRAIN-B	OR	DG-B, dummyGate2
dummyGate2	OR	C-PUMP-B, C-MOV-B, C-CV-B
1987/12/15	16:24:07	page 1

Figure 97. Fault Tree Modified Logic Report for the CCS fault tree.

Notice that the modified logic contains gates not defined in the original logic. IRRAS 2.0 creates new gates in the process of coalescing and separating out independent subtrees and names these

gates "dummyG~ where n is the next available number in sequence.

3.8.~ Fault Tree Importance Measures Report. The next fault tr report is the Fault Tree Importance Measures Report. It is obtained by selecting option I - Importance Measures on the Fault Tree Reports menu in Figure 94, highlighting the desired fault tree, and pressing <Enter>. A title and output directions are indicated as described in Section 3.8.1.1. Figure 98 shows what this report looks like.

FAULT TREE IMPORTANCE MEASURES REPORT				
Family: DEMO FAMILY				
Fault Tree: CCS				
(Sorted by Fussell-Vesely)				
Event Name	Probability of Failure	Fussell- Vesely Importance	Risk Reduction Ratio	Risk Increase Ratio
DG-B	2.000E-002	9.421E-001	1.727E+001	4.716E+001
C-MOV-1	1.000E-003	4.621E-002	1.048E+000	4.716E+001
DG-A	2.000E-002	7.479E-003	1.008E+000	1.366E+000
C-MOV-B	5.000E-003	6.487E-003	1.007E+000	2.283E+000
C-PUMP-B	3.000E-003	3.892E-003	1.004E+000	2.285E+000
C-MOV-A	5.000E-003	1.870E-003	1.002E+000	1.371E+000
C-PUMP-A	3.000E-003	1.122E-003	1.001E+000	1.372E+000
C-CV-B	1.000E-004	1.297E-004	1.000E+000	2.289E+000
C-CV-A	1.000E-004	3.739E-005	1.000E+000	1.373E+000
TANK	1.000E-007	4.616E-006	1.000E+000	4.716E+001
1987/12/15	16:25:33	page	1	

Figure 98. Fault Tree Importance Measures Report.

This report contains:

- o Header. The header consists of the report title, the family name, the fault tree name, and a note indicating that the list of events are sorted by the Fussell-Vesely Importance. The header is at the top of each page of the report.
- o Event Name. This column lists every basic event in the fault tree.

- o Probability of Failure. The basic event failure probability calculated by IRRAS 2.0 using the input calculation type and failure data.
- o Fussell-Vesely Importance. The Fussell-Vesely Importance of the basic event to the failure of the fault tree top event. For more on the importance measures, see Sections 3.5.3 and 6.5.
- o Risk Reduction Ratio. The risk reduction ratio for the basic event with respect to the fault tree top event (see Sections 3.5.3 and 6.5).
- o Risk Increase Ratio. The risk increase ratio for the basic event with respect to the fault tree top event (see Sections 3.5.3 and Section 6.5).
- o Footer. The footer at the bottom of each page contains the date and time that the report was generated and the page number.

The Fussell-Vesely importance, Risk Reduction Ratio, and Risk Increase Ratio came about from choosing ratios on the IRRAS Constants menu (Figure 10) when the question was asked, Use ratios in Importance Measures?. If interval had been chosen, the values displayed would be the Birnbaum importance measure, risk reduction interval, and risk increase interval. For detailed discussion of importance measures, see Section 6.5.

3.8.2.5 Fault Tree Cut Sets Report. The fourth fault tree report available on the Fault Tree Reports menu (Figure 94) is the Fault Tree Cut Sets Report. This report is obtained by selecting option C - Cut sets, highlighting the desired fault tree, and pressing <Enter>. IRRAS 2.0 displays the question Cut Set Report Options list and places a cursor on the answer field. The default answer is Y, yes. This is shown in Figure 99.

I R R A S R E P O R T S
Fault Tree Reports

Currently Selected Family => DEMO FAMILY

Cut Set Report Options

Please specify the following information to be used in generating the selected Cut Set report:

Total number of Cut Sets 9999
Max size of each Cut Set 99

Quantified Report (Y/N)? Y
Quantified value cutoff +0.0E+000
Percent of total value 100.0

Figure 99. IRRAS Fault Tree Reports menu with Cut Sets Report Options.

Answering yes to the Quantified Report question will generate a cut set report with the cut sets sorted according to probability. Answering no gives the same report without any probability information. After supplying a title and output directions, the Fault Tree Cut Sets with Quantification Report looks like Figure 100.

FAULT TREE CUT SETS (QUANTIFICATION) REPORT				
Family: DEMO FAMILY				
Fault Tree: CCS				
Mincut Upper Bound 2.120E-002				
Cut No.	% Total	% Cut Set	Freq.	Cut Sets
1	94.3	94.3	2.0E-002	DG-B
2	99.1	4.7	1.0E-003	C-MOV-1
3	99.5	.5	1.0E-004	C-MOV-B, DG-A
4	99.8	.3	6.0E-005	C-PUMP-B, DG-A
5	99.9	.1	2.5E-005	C-MOV-A, C-MOV-B
6	100.0	.1	1.5E-005	C-MOV-A, C-PUMP-B
7	100.0	.1	1.5E-005	C-MOV-B, C-PUMP-A
8	100.0	.0	9.0E-006	C-PUMP-A, C-PUMP-B
9	100.0	.0	2.0E-006	C-CV-B, DG-A
10	100.0	.0	5.0E-007	C-CV-A, C-MOV-B
11	100.0	.0	5.0E-007	C-CV-B, C-MOV-A
12	100.0	.0	3.0E-007	C-CV-A, C-PUMP-B
13	100.0	.0	3.0E-007	C-CV-B, C-PUMP-A
14	100.0	.0	1.0E-007	TANK
15	100.0	.0	1.0E-008	C-CV-A, C-CV-B
1987/12/15 16:26:42 page 1				

Figure 100. Fault Tree Cut Sets with Quantification Report.

This report contains:

- o Header. The header consists of the report title, the family name, the fault tree name, and the Minimal Cut Set Upper Bound value. This information is repeated at the top of every page of the report.
- o Cut No. The ranking of the cut set according to frequency.
- o % Total. The cumulative percentage of the cut sets to the minimal cut set upper bound.
- o % Cut set. The percentage contribution of the cut set to the minimal cut set upper bound.
- o Frequency. The probability of the cut set occurring.

- o Cut sets. The basic events that make up each cut set.
- o Footer. The footer at the bottom of each page contains the date and time that the report was generated and the page number.

The percentage calculations are only approximations due to the way the minimal cut set upper bound is calculated from the cut set probabilities. Since the minimal cut set upper bound is not just a simple summing of the cut set probabilities, the ratio of the cut set probability to the minimal cut set upper bound is slightly greater than the true percentage; therefore, the total of the percentages is greater than 100%. IRRAS 2.0 truncates the % total at 100%. For more on the calculation of the minimal cut set upper bound and the percentages, see Section 6.2.

3.8.2.6 Fault Tree Summary Report. The fifth fault tree report listed on the Fault Tree Reports menu (Figure 94) is the Fault Tree Summary Report. It is obtained by selecting the S - Summary option and pressing <Enter>. This report does not require that a fault tree be highlighted, since there is only one fault tree summary report per family. A title and output directions are supplied on the IRRAS Reports Information Request form (Figure 90). The Fault Tree Summary Report is shown in Figure 101.

FAULT TREE SUMMARY REPORT			
Family: DEMO FAMILY			
Tree No.	Fault Tree Name	Level	MinCut Upper Bound
1	CCS Containment Cooling System fault tree	0	2.120E-002
2	ECS Emergency Cooling System fault tree	0	2.120E-002
3	LOSP Loss of Offsite Power Initiating Event fault tree	0	2.300E+000
1987/12/15		16:27:23	page 1

Figure 101. Fault Tree Summary Report.

This report contains:

- o Header. The header consists of the report title and the family name. This information is repeated at the top of every page of the report.
- o Tree No. The numerical order that the trees were loaded into the data base.
- o Fault Tree Name. The name used to save the fault tree in the data base.
- o Level. The level assignment of the fault tree. See Section 3.3.6 for more on fault tree levels.
- o MinCut Upper Bound. The calculated minimal cut set upper bound value for the fault tree as determined by the point estimate quantification.
- o Footer. The footer at the bottom of each page contains the date and time that the report was generated and the page number.

3.8.2.7 Fault Tree Uncertainty Values Report. The last fault tree report listed on the Fault Tree Reports menu (Figure 94) is the Fault Tree Uncertainty Values Report. It is obtained by selecting the U - Uncertainty Values option and pressing <Enter>. This report does not require that a fault tree be highlighted, since there is only one fault tree Uncertainty Values report per family. A title and output directions are supplied on the IRRAS Reports Information Request form (Figure 90). The Fault Tree Uncertainty Values Report is shown in Figure 102.

FAULT TREE UNCERTAINTY VALUES REPORT							
Family: DEMO FAMILY							
Tree No.	Fault Tree Name	Random Seed = 3571 Sample Size = 1000 Uncertainty Values					
1	CCS	Mean	2.061E-002	5th	1.542E-003	Min.	3.669E-004
		MinCut	2.120E-002	Median	9.407E-003	Max.	3.933E-001
				95th	7.255E-002	St.D.	3.776E-002
2	ECS	Mean	2.061E-002	5th	1.542E-003	Min.	3.669E-004
		MinCut	2.120E-002	Median	9.407E-003	Max.	3.933E-001
				95th	7.255E-002	St.D.	3.776E-002
3	LOSP	Mean	2.508E+000	5th	9.267E-002	Min.	5.228E-003
		MinCut	2.300E+000	Median	7.969E-001	Max.	1.117E+002
				95th	8.676E+000	St.D.	6.782E+000
1988/03/01		10:17:27			page 1		

Figure 102. Fault Tree Uncertainty Values Report.

This report contains:

- o Header. The header consists of the report title and the family name. This information is repeated at the top of every page of the report.
- o Tree No. The numerical order that the trees were loaded into the data base.
- o Fault Tree Name. The name used to save the fault tree in the data base.
- o Uncertainty Values. Summary results from the uncertainty analyses, including the 5th, median, mean, and 95th percentile values of the distribution; the minimum and maximum sample results; and the standard deviation. Also included is the point estimate minimal cut set upper bound value.
- o Footer. The footer at the bottom of each page contains the date and time that the report was generated and the page number.

3.8.3 Sequence Reports

To access the reports on the sequences, we selected option S - Sequences, on the IRRAS Reports menu (Figure 87), highlighted the DEMO FAMILY, and pressed <Enter>. IRRAS 2.0 put the Sequence Reports menu on the screen. This menu is shown in Figure 103.

I R R A S R E P O R T S
Sequence Reports

Currently Selected Family => DEMO FAMILY

Option |E| Exit / Logic / Summary / Uncertainty Values <= other options
 Importance Measures / Cut Sets <= choose a sequence

S E Q U E N C E S

1-LOSP-ECS	Loss of Offsite Power with success of ECS, no core damage.
2-LOSP-ECS-CCS	Loss of Offsite Power, failed ECS, successful CCS, sm. rel.
3-LOSP-ECS-CCS	Loss of Offsite Power, failed ECS, failed CCS, lg. release.

Note : Use <PgUp> and <PgDn> to display more sequences.

Figure 103. Sequence Reports menu.

This menu lists the sequence reports available in IRRAS 2.0. These reports are described in the sections that follow.

3.8.3.1 Sequence Logic Report. The Sequence Logic Report displays the combination of failed and successful fault trees and the initiating event that constitutes the sequence. This is the sequence logic used as input to the SEQUENCE Analysis module.

The report was obtained by selecting option L - Logic from the Sequence Reports menu and pressing <Enter>. This report does not require that a sequence be highlighted. The IRRAS Reports Information Request form appeared on the screen. This form was used to give the report a title and to tell IRRAS 2.0 whether to send the report to the screen, printer, or file on the hard disk. The report is shown in Figure 104.

SEQUENCE LOGIC REPORT			
Family: DEMO FAMILY			
Seq. No.	Sequence Name	Logic	
1	1-LOSP-ECS	LOSP	/ ECS
2	2-LOSP-ECS-CCS	ECS	LOSP / CCS
3	3-LOSP-ECS-CCS	CCS	ECS LOSP
1987/12/18		14:19:59	page 1

Figure 104. Sequence Logic Report.

This report contains:

- o Header. The header consists of the report title and the family name. This information is repeated at the top of every page of the report.
- o Seq. No. The order that the sequences were entered into the data base.
- o Sequence Name. The name used to save the sequence in the data base.
- o Sequence Logic. The initiating event and an indication of the successful and failed systems in the sequence. See Section 3.6 for more information on sequence logic.
- o Footer. The footer at the bottom of each page contains the date and time that the report was generated and the page number.

3.8.3.2 Sequence Summary Report. The second fault tree report listed on the Sequence Reports menu (Figure 103) is the Sequence Summary Report. It is obtained by selecting the S - Summary option and pressing <Enter>. This report does not require that a sequence be highlighted since there is only one sequence summary report per family. A title and output directions are

supplied on the IRRAS Reports Information Request form (Figure 90). The Sequence Summary Report is shown in Figure 105.

SEQUENCE SUMMARY REPORT		
Family: DEMO FAMILY		
Seq No.	Sequence Name	MinCut Upper Bound
1	1-LOSP-ECS Loss of Offsite Power with success of ECS, no core damage.	2.300E+000
2	2-LOSP-ECS-CCS Loss of Offsite Power, failed ECS, successful CCS, sm. rel.	4.834E-002
3	3-LOSP-ECS-CCS Loss of Offsite Power, failed ECS, failed CCS, lg. release.	1.759E-003
	TOTAL	2.350E+000
1988/03/01 10:18:29 page 1		

Figure 105. Sequence Summary Report.

This report contains:

- o Header. The header consists of the report title and the family name. This information is repeated at the top of every page of the report.
- o Seq. No. The numerical order that the sequences were loaded into the data base.
- o Sequence Name. The name used to save the sequence in the data base.
- o Sequence Description. The 60-character description entered in the data base for each sequence.
- o MinCut Upper Bound. The calculated minimal cut set upper bound value for the sequence as determined by the point estimate quantification.

- o Footer. The footer at the bottom of each page contains the date and time that the report was generated and the page number.

3.8.3.3 Sequence Uncertainty Values Report. The third sequence report listed on the Sequence Reports menu (Figure 103) is the Sequence Uncertainty Values Report. It is obtained by selecting the U - Uncertainty Values option and pressing <Enter>. This report does not require that a sequence be highlighted, since there is only one Sequence Uncertainty Values report per family. A title and output directions are supplied on the IRRAS Reports Information Request form (Figure 90). The Sequence Uncertainty Values Report is shown in Figure 106.

SEQUENCE UNCERTAINTY VALUES REPORT							
Family: DEMO FAMILY							
Seq. No.	Sequence Name	Random Seed = 3571 Sample size = 1000 Uncertainty Values					
1	1-LOSP-ECS	Mean	2.508E+000	5th	9.267E-002	Min.	5.228E-003
		MinCut	2.300E+000	Median	7.969E-001	Max.	1.117E+002
				95th	8.676E+000	St.D.	6.782E+000
2	2-LOSP-ECS-CCS	Mean	5.244E-002	5th	4.097E-004	Min.	1.099E-005
		MinCut	4.834E-002	Median	8.104E-003	Max.	6.741E+000
				95th	1.830E-001	St.D.	2.550E-001
3	3-LOSP-ECS-CCS	Mean	4.568E-003	5th	4.373E-006	Min.	1.380E-007
		MinCut	1.759E-003	Median	1.817E-004	Max.	6.112E-001
				95th	8.723E-003	St.D.	3.541E-002
1988/03/01		10:18:46			page		1

Figure 106. Sequence Uncertainty Values Report.

This report contains:

- o Header. The header consists of the report title and the family name. This information is repeated at the top of every page of the report.
- o Seq. No. The numerical order that the sequences were loaded into the data base.

- o Sequence Name. The name used to save the sequence in the data base.
- o Uncertainty Values. Summary results from the uncertainty analyses, including the 5th, median, mean, and 95th percentile values of the distribution; the minimum and maximum sample results; and the standard deviation. Also included is the point estimate minimal cut set upper bound value.
- o Footer. The footer at the bottom of each page contains the date and time that the report was generated and the page number.

3.8.3.4 Sequence Importance Measures Report. The next sequence report is the Sequence Importance Measures Report. It is obtained by selecting option I - Importance Measures on the Sequence Reports menu in Figure 103, highlighting the desired sequence, and pressing <Enter>. A title and output directions are indicated as described in Section 3.8.1.1. Figure 107 shows what this report looks like.

SEQUENCE IMPORTANCE MEASURES REPORT				
Family: DEMO FAMILY				
Sequence: 2-LOSP-ECS-CCS				
(Sorted by Fussell-Vesely)				
Event Name	Probability of Failure	Fussell- Vesely Importance	Risk Reduction Ratio	Risk Increase Ratio
IE-LOSP	2.300E+000	1.000E+000	4.143E+014	4.354E-001
DG-A	2.000E-002	9.493E-001	1.973E+001	4.752E+001
E-MOV-1	1.000E-003	4.539E-002	1.048E+000	4.634E+001
E-MOV-A	5.000E-003	1.834E-003	1.002E+000	1.363E+000
E-MOV-B	5.000E-003	1.834E-003	1.002E+000	1.363E+000
E-PUMP-B	3.000E-003	1.100E-003	1.001E+000	1.364E+000
E-PUMP-A	3.000E-003	1.100E-003	1.001E+000	1.364E+000
E-CV-B	1.000E-004	3.668E-005	1.000E+000	1.365E+000
E-CV-A	1.000E-004	3.668E-005	1.000E+000	1.365E+000
1988/03/01	10:18:00	page	1	

Figure 107. Sequence Importance Measures Report.

This report contains:

- o Header. The header consists of the report title, the family name, the sequence name, and a note indicating that the list of events are sorted by the Fussell-Vesely Importance. The header is at the top of each page of the report.
- o Event Name. This column lists every basic event in the sequence.
- o Probability of Failure. The basic event failure probability calculated by IRRAS 2.0 using the input calculation type and failure data.
- o Fussell-Vesely Importance. The Fussell-Vesely importance of the basic event to the sequence probability. For more on the importance measures, see Sections 3.7.3 and 5.5.
- o Risk Reduction Ratio. The risk reduction ratio for the basic event with respect to the sequence probability (see Sections 3.7.3 and 5.5).
- o Risk Increase Ratio. The risk increase ratio for the basic event with respect to the sequence probability (see Section 3.7.3 and Section 5.5).
- o Footer. The footer at the bottom of each page contains the date and time that the report was generated and the page number.

3.8.3.5 Sequence Cut Sets Report. The last sequence report available on the Sequence Reports menu (Figure 103) is the Sequence Cut Sets Report. This report is obtained by selecting option C - Cut sets, highlighting the desired sequence, and pressing <Enter>. IRRAS 2.0 displays IRRAS Sequence Report Cut Set Report Option shown in Figure 108. In this menu, the user can specify various output options for the report.

I R R A S R E P O R T S
Fault Tree Reports

Currently Selected Family => DEMO FAMILY

— Cut Set Report Options —

Please specify the following information to be used in generating the selected Cut Set report:

Total number of Cut Sets 9999
Max size of each Cut Set 99

Quantified Report (Y/N)? Y
Quantified value cutoff +0.0E+000
Percent of total value 100.0

Figure 108. Sequence Reports menu for the Sequence Cut Sets Report.

After supplying a title and output directions, the Sequence Cut Sets with Quantification Report looks like Figure 109.

SEQUENCE CUT SETS (QUANTIFICATION) REPORT					
Family: DEMO FAMILY					
Sequence: 2-LOSP*ECS*/CCS					
Mincut Upper Bound 4.834E-002					
Cut No.	% Total	% Cut Set	Freq.	Cut Sets	
1	95.2	95.2	4.6E-002	DG-A, IE-LOSP	
2	99.9	4.8	2.3E-003	E-MOV-1, IE-LOSP	
3	100.0	.1	5.7E-005	E-MOV-A, E-MOV-B, IE-LOSP	
4	100.0	.1	3.5E-005	E-MOV-A, E-PUMP-B, IE-LOSP	
5	100.0	.1	3.5E-005	E-MOV-B, E-PUMP-A, IE-LOSP	
6	100.0	.0	2.1E-005	E-PUMP-A, E-PUMP-B, IE-LOSP	
7	100.0	.0	1.1E-006	E-CV-B, E-MOV-A, IE-LOSP	
8	100.0	.0	1.1E-006	E-CV-A, E-MOV-B, IE-LOSP	
9	100.0	.0	6.9E-007	E-CV-A, E-PUMP-B, IE-LOSP	
10	100.0	.0	6.9E-007	E-CV-B, E-PUMP-A, IE-LOSP	
11	100.0	.0	2.3E-008	E-CV-A, E-CV-B, IE-LOSP	
1988/03/01			11:03:09		page 1

Figure 109. Sequence Cut Sets with Quantification Report.

This report contains:

- o Header. The header consists of the report title, the family name, the sequence name, and the Mincut Upper Bound value. This information is repeated at the top of every page of the report.
- o Cut No. The ranking of the cut set according to frequency.
- o % Total. The cumulative percentage of the cut sets to the minimal cut set upper bound.
- o % Cut Set. The percentage contribution of the cut set to the minimal cut set upper bound.
- o Frequency. The probability of the cut set occurring.
- o Cut Sets. The basic events that make up each cut set.

- o Footer. The footer at the bottom of each page contains the date and time that the report was generated and the page number.

The percentage calculations are only approximations due to the way the minimal cut set upper bound is calculated from the cut set probabilities. Since the minimal cut set upper bound is not just a simple summing of the cut set probabilities, the ratio of the cut set probability to the minimal cut set upper bound is slightly greater than the true percentage; therefore, the total of the percentages is greater than 100%. IRRAS 2.0 truncates the % Total at 100%. For more on the calculation of the minimal cut set upper bound and the percentages, see Section 6.2.

3.8.4 Graphics Printouts

IRRAS 2.0 has the ability to generate report-quality graphics by sending the fault tree drawings as shown on the screen in the Build Fault Trees module to a laser printer or a six-color pen plotter. Draft (not report quality) graphics can be obtained by sending the drawings to a dot matrix printer.

3.8.4.1 Graphics on a Dot Matrix Printer. The graphics created in the BUILD Fault Trees module can be sent to an Epson FX+ series printer. This is done from within the BUILD Fault Trees module. To send a drawing to the printer, we entered the BUILD Graphic Trees module from the Fault Tree Graphics System menu, called up the desired drawing on the screen and selected the FILE submenu from the main graphics menu. On this submenu, we selected the PLOT EPS option with the left button on the mouse. The cursor went away; IRRAS 2.0 redrew the picture in black and white and then started the printer. The printer drew the picture one line at a time. When the printer was finished, IRRAS 2.0 redrew the picture on the screen in color and gave back the cursor, ready for the next command.

3.8.4.2 Graphics on a Laser Printer. Report-quality graphics are available from IRRAS 2.0 if a high-resolution laser printer, such as the HP Laserjet, is used. To use a laser printer, the graphics in IRRAS 2.0 must be converted to a form acceptable by the printer. IRRAS 2.0 performs this conversion, called rasterization, in the BUILD Fault Trees module. All of the graphics in this manual have been printed out using this method.

NOTE: In future versions of IRRAS, a Post Script compatible driver will be implemented to allow the user to generate plots directly on printers

supporting the Post Script graphical format. This will give the user better quality plots without the need to go through the rasterization step.

To rasterize a fault tree, we entered the BUILD Fault Tree module from the IRRAS 2.0 main menu. This brought up the IRRAS Fault Tree Graphics System menu shown in Figure 110.

Figure 110. IRRAS-PC Fault Tree Graphics System menu (option R).

From this menu, we chose option R - Rasterize Tree. This called up the Rasterize Fault Trees menu displayed in Figure 111.

R a s t e r i z e F a u l t T r e e s

Option |R| Exit / Rasterize
 Low / High

<-- select a Fault Tree below
<-- LOW resolution selected

CCS ECS LOSP
-----FAULT TREES-----

Note : Use <PgUp> and <PgDn> to display more Fault Trees.
If you do not select a Fault Tree, all will be done.

Figure 111. Rasterize Fault Trees menu.

When this menu came up, the R - Rasterize option was already selected and the resolution was set to Low or 150 dots per inch. We could have selected the High resolution mode which would have given a 300-dots-per-inch quality at the expense of time and file space. We used the arrow keys to move the cursor through the list of fault trees to the tree that we wanted to rasterize and then pressed <Enter>. The cursor moved to the lower left corner of the screen and the message

Rasterizing fault tree => CCS.DLS

was displayed at the bottom of the screen. Since this process converts every pixel on the screen, rasterizing takes some time. For an XT, it can take about five or six minutes. During this time, information is written to the hard disk and the cursor blinks, indicating that the process is in fact working and the machine is not locked up. The rasterization process creates a file in the current directory of 152,353 bytes for each fault tree rasterized (low resolution), 555,553 bytes for the high-resolution format. The file name is the fault tree name with a suffix of .RAS. When IRRAS 2.0 completes the rasterization, it displays a message stating

Raster image created - file is => CCS.RAS

and the cursor returns to the highlighted fault tree. From here we can select another fault tree to rasterize, or we can move the cursor up to the option block and exit.

NOTE: Because the rasterizing process creates fairly large files, it is easy to fill up the hard disk if there are a number of files to process. Should the computer run out of disk space, IRRAS 2.0 will no longer be able to rasterize files or store more data. This will most likely result in the computer locking up, requiring the user to power down the system and start afresh. It is best to know how much room is available on the hard disk and plan accordingly.

A batch option is available for rasterizing fault tree files. This option is invoked by selecting R - Rasterize <Enter> without highlighting a fault tree. A warning message will appear asking if the user desires to continue. To continue in the batch mode, press Y <Enter>.

To get the printout from the laser printer, we exited IRRAS 2.0 to the DOS prompt in the current directory. In this directory, we found the raster files created with the Rasterize Tree option. We sent these files to the printer one at a time by typing in the command

```
copy/b treename.ras printername <Enter>
```

For example, to print out the CCS fault tree on our laser printer designated lpt1, we typed

```
copy/b ccs.ras lpt1 <Enter>
```

The file was copied to the printer, the ready light on the printer indicated that the printer was in use, and after 30 seconds to a minute, a printout of our fault tree was obtained. The fault tree drawings in this manual are examples of the output of this process.

NOTE: Due to the size of the raster file, the laser printer may not be able to properly print out the fault tree drawing if character fonts are loaded. If only a partial drawing

is obtained from the printer, set up the printer without the character fonts and re-send the raster file to the printer.

3.8.4.3 Color Graphics on a Pen Plotter. IRRAS 2.0 was developed to work with the HP 7475 six-pen plotter. The pen plotter provides the user with color graphics. There are three basic steps to obtaining color graphics. These are

- o mapping the screen colors to the plotter pen
- o creating a file compatible with the pen plotter
- o plotting the fault tree.

The first step, mapping the screen colors to the plotter pens, was done by selecting the BUILD Fault Trees option from the IRRAS 2.0 main menu. From the IRRAS Fault Tree Graphics System menu (see Figure 110), we chose the D - Define plotter pens option. This brought up a pen color map showing the 16 colors available in IRRAS 2.0, the current pen number assignments, and a stop sign. Using the mouse, we moved the cursor to the first color block and pressed the right button. A blue box was drawn around the chosen color and the prompt at the top of the screen read

Enter number of pen >

The allowed pen numbers in IRRAS 2.0 are 1 through 6, corresponding to the numbers on the plotter pens. We entered the desired pen number and pressed <Enter>. The number beside the chosen color square was updated and the prompt changed to

Pick color to be mapped or pick STOP

We continued until all the colors were mapped as we desired. Then we picked the STOP sign to exit this menu.

NOTE: More than one color must be mapped to some pens, since there are 16 colors and only 6 pens. The color mapping will remain in effect for all plots until changed as described above.

The next step was to create files to be sent to the plotter. This was done by going into the Build Fault Trees option from the IRRAS Fault Tree Graphics System menu. Using the mouse, we called up the desired fault tree. Then on the FILE submenu, we chose the

PLOT HP option with the left button. The prompt at the bottom of the screen read

Enter File name or CR for file CCS.HPP>

The screen went blank and the fault tree was redrawn. When this was completed, the plotter file was created. The cursor was returned, ready for another command. The file created was called treename.hpp. For example, the plot file for the CCS fault tree was called

ccs.hpp

The final step was to plot the drawing. This required telling IRRAS 2.0 to which communications port the plotter was connected. We did this by going into the UTILITY Options module on the IRRAS 2.0 Main menu. This brought up the Utilities menu. On this menu, we chose the D - Define Constants option. The IRRAS Constants User Information form appeared on the screen, as shown in in Figure 112. We then moved the highlight to the plotter communications port and entered the number of the port to which our plotter was connected. When the desired changes are made, hit <Enter>. This brought up the IRRAS Constants Graphic Information menu. Although IRRAS 2.0 brings up this menu, no changes are required. Strike <Enter> to bring back the Utilities menu. From the Utilities menu, we exit back to the IRRAS 2.0 main menu.

I R R A S C o n s t a n t s			
-----User Information-----			
User Name	EG&G Idaho, Inc. (INEL)		
Monitor type (0-EGA, 1-CGA)			0
Locator type (1-Keyboard/Tablet/Logitech, 2-Microsoft)			1
Printer has complete IBM character font? (Y/N)			N
Plotter communications port (0-None, 1-Com1, 2-Com2, 3-Com3)			0
Use primary basic event Names ? (Y=primary, N=alternate)			Y
Use ratios in Importance Measures ? (Y=ratios, N=intervals)			Y
-----Uncertainty-----			
Random number seed for uncertainty calculations			3571
Sample size			1000
-----Cut Set Generation-----			
Cutoff by size? (Y/N)	Y	Size cutoff	6
Cutoff by probability? (Y/N)	Y	Probability cutoff	1.000E-015
Mission time (hours)			2.400E+001
-----File Locations-----			
Halo drive and directory path	\HALO\		

Change any of the constant values shown and press the <Enter> key.

Figure 112. IRRAS Constants form for plotter communication port.

Next, we selected Build Fault Trees option on the IRRAS 2.0 main menu. This brought us to the Fault Tree Graphics System menu (Figure 110) where we selected the P-Plot Tree (plotter) option. The PLOT H.P. Fault Tree menu appeared, as shown in Figure 113. Choose the file desired using the arrow keys and hit <Enter>. The system told us to

Press <ENTER> when plotter is ready

We set up the plotter and pressed <Enter>. The Halo Plotter Driver copyright message came on the screen, and the plotter began drawing. When the drawing was finished, the copyright message went away and IRRAS 2.0 displayed the

All plots completed

message. The system was ready for our next command.

NOTE: If more than one treename.hpp file is in the current directory when the Plot option is executed, IRRAS 2.0 will plot one fault tree after another. IRRAS 2.0 stops between each plot to allow the user to change paper. When the plotter is set up for the next drawing, the user presses <Enter>. Failure to change the plotter paper will result in drawings being drawn on top of one another.

NOTE: Executing the P - Plot Tree option without having a plotter connected to your system will bring up the computer message:

No Plotter Attached

This will happen provided the plotter communications port has the default of 0 in the IRRAS Constants form (see Figure 112). If any other communications port is identified without having a plotter connected to your system, the system will hang up. Reboot the system to regain control. In some cases, it may be necessary to turn off the computer and start fresh.

Plot H. P. Fault Trees

Option |R| Exit / Plot / Delete ← select a Plot File below

PLOT TREES

CCS

Note : Use <PgUp> and <PgDn> to display more Plot files.
If you do not select a Fault Tree, all will be done.

Figure 113. Plot H.P. Fault Trees menu.

4. ADDITIONAL CAPABILITIES OF IRRAS 2.0

4.1 More on the Building of Fault Trees

The BUILD Fault Trees module graphically builds fault tree diagrams and defines which fault tree shapes are related to other shapes. The BUILD Fault Trees module is accessed from the IRRAS 2.0 main menu. Figure 114 shows the IRRAS-PC Fault Tree Graphics System menu.

Figure 114. IRRAS-PC Fault Tree Graphics System menu.

4.1.1 Clearing Extracted Fault Trees

The graphic fault trees may be stored in two places by IRRAS 2.0. The first place is within the IRRAS 2.0 data base. Every fault tree drawing is automatically put in the data base when the graphics building module is exited. IRRAS 2.0 displays messages indicating the successful loading of each fault tree drawing into the data base and then indicates when the data base loading is completed.

IRRAS 2.0 also creates a file in the current directory for each fault tree created. Each file is named

treename.DLS

where treename is the name assigned during the SAVE TREE filing process (see Sections 3.3.5.3 and 3.3.5.6).

To save space and keep the data base directory clean, it is usually desirable to remove the .DLS files. This can be done within the IRRAS 2.0 system by selecting the X - eXtract Fault Trees option shown on Figure 114. This displays the Extract Fault Trees menu shown in Figure 115.

Extract Fault Trees

Currently Selected Family —> DEMO FAMILY

Option |E| Exit / eXtract Tree / Clear extracted Trees /Display extracted Trees

FAULT TREES

CCS	Containment Cooling System fault tree
ECS	Emergency Cooling System fault tree
LOSP	Loss of Offsite Power Initiating Event fault tree

Note : Use <PgUp> and <PgDn> to display more fault trees.

Figure 115. Extract Fault Trees Menu.

This menu lists all the fault trees in the selected family and provides options to extract fault trees, clear extracted fault trees, and display the extracted fault trees. To remove the .DLS files from the current directory, select C - Clear extracted Trees and press <Enter>. IRRAS 2.0 will show the following messages at the bottom of the screen:

Clearing extracted files...please wait

and

All extracted files cleared...

This indicates that all .DLS files have been removed from the current directory and are stored only within the IRRAS 2.0 data base.

NOTE: The Clear extracted Trees option clears ALL extracted fault trees from the current directory,

even those for fault trees in another family. Other fault tree files (such as .HPP and .SET) are not affected. Files in other directories or subdirectories are also not affected.

4.1.2 Extracting Fault Trees from the Data Base

To speed up the graphics portion of IRRAS 2.0, the Build Graphics Trees option (Figure 114) accesses only the fault tree drawings that have been extracted from the data base. The LIST TREES and LOAD TREE options within the graphics (see Section 3.3.5.4) operate only on the extracted fault trees. Creation of a new fault tree drawing does not require extraction before entering the graphics module.

NOTE: To access the graphics for a fault tree, it must first be extracted from the data base. All other operations associated with fault trees (analysis, rasterization, etc.) do not require the trees to be extracted.

Extracting fault trees is done by selecting the eXtract Tree option on the eXtract Fault Trees menu. To do so, press X, highlight the fault tree desired, and press <Enter>. IRRAS 2.0 displays

Extracting fault tree - please wait...

at the bottom of the screen. This is followed by

Fault Tree Extraction complete...

This process is repeated until all the desired fault trees in the current family are extracted (.DLS files created). To display all the fault trees that are extracted, press D for Display extracted Trees and press <Enter>. The display of Figure 116 will appear on the screen.

Figure 116. Display of extracted fault trees.

This display shows all the extracted fault trees for the current directory regardless of which family is currently selected.

4.1.3 Build Graphic Trees

The Build Graphics Trees option on the IRRAS-PC Fault Tree Graphics System menu (Figure 114) is menu-driven with options selected using a mouse, graphic tablet puck, or cursor control keys on the keyboard. The graphics module is exited by picking EXIT, which is on the main graphics menu and the FILE submenu.

The graphics module contains:

1. Predefined shapes for constructing fault tree diagrams.
2. Edit functions for moving, copying, and deleting shapes, lines, and text.
3. Edit functions for changing attributes.
4. Assignment of names to shapes.
5. Descriptive text.
6. A file system for storing and retrieving drawings and for paging between drawings.

7. Output to the Hewlett-Packard pen plotter, Hewlett-Packard Laser Jet II laser printer, and Epson printer/plotter.
8. Construction of tabular relational data from the graphical diagrams.

4.1.3.1 Basic Operation of Menus. Options in the graphics module are selected using the screen cursor. The cursor can be operated using a mouse, tablet puck, or keys on the keyboard. The mouse or tablet puck must have at least two buttons. The buttons put up and take down menus, make command selections, and locate objects. In describing the graphics module, it is assumed that the cursor control device is a LOGITECH three-button mouse. If a two-button mouse is used, the right button corresponds to the right button on the three-button mouse. If the keyboard is used, keys "\" and "z" correspond to the right button and the left shift corresponds to the left button on the LOGITECH mouse. If a Summagraphics tablet is used with a four button puck, the top and right buttons correspond to the right and left buttons on the LOGITECH mouse. If another cursor locator device is used, the position (purpose) of the buttons on the device may be different.

When initially entering the graphics module by way of the Build Graphics Trees option on the Fault Tree Graphics System menu the screen is blank with an attribute list at the top. The main graphics menu is put up by moving the cursor to the desired location and pressing the left button on the mouse. The main graphics menu will be displayed at that location, as illustrated in Figure 117. To move the main menu, move the screen cursor to another location (outside of the menu) and press the left button. The menu will move to the new location. It can be taken down by putting the cursor on the menu and pressing the right-button on the mouse. Pop-up menus, as opposed to fixed menus, were chosen for two reasons. First, pop-up menus can be placed near the location of drawing or moved out of the way of a section of drawing. Second, and more important, the pop-up menus free the entire screen for drawing.

Figure 117. Graphics module primary menu.

Submenus are displayed by picking the submenu name from the primary menu using the left button on the mouse. The submenu is drawn on top of the primary menu with the upper-left corner of the submenu at the location of the cursor. Figure 118 illustrates the EDIT submenu. Items can be picked from any menu shown. For example, in Figure 118, the COPY command from the EDIT menu may be picked as well as the SHOW command from the main graphics menu. The top menu (in this case the EDIT menu) has preference when menus overlay one another.

Figure 118. EDIT submenu overlaying main graphics menu.

A menu can be taken down by pointing at the menu and pressing the right button on the mouse. If other menus are sitting on top of the menu being taken down, those menus also are taken down.

The general rules for using the mouse buttons are:

1. The left button puts up the main menu or picks a command or submenu from a menu.
2. The right button takes down a menu or terminates a multiple-pick command.

4.1.3.2 Basic Fault Tree Symbols and Shapes. The fault trees are built using predefined shapes, connecting lines, and text. The available predefined shapes are listed on the BUILD submenu, shown in Figure 119. The shapes are illustrated in Figure 120. A shape (such as an OR gate) is chosen from the menu using the screen cursor. A ghost (bitmap) of the shape is generated and can be moved about the screen. When located at the proper location, the ghost shape is permanently placed by picking the left button on the mouse, and another shape or command can be picked. Lines connect shapes together.

Figure 119. The BUILD submenu.

Figure 120. Fault tree basic shapes.

The shapes available on the BUILD submenu are described as follows: (Portions of the descriptions below have been taken from Reference 2.)

- o LINE. Lines are used to connect the shapes to create the fault tree logic. IRRAS 2.0 provides solid lines, dashed lines, and dotted lines for display purposes. All line types are processed as solid lines throughout IRRAS 2.0.
- o AND gate. Output fault occurs if all of the input faults occur.
- o OR gate. Output fault occurs if at least one of the input faults occurs.
- o BEVENT. The basic event shape indicates that the fault requires no further development.
- o TABLE. Input faults are listed in tabular form. Up to 8 basic events can be listed.

- o N/M OR gate. Output fault occurs if any combination of N faults occurs.
- o UEVENT. An undeveloped event indicating that the logic for the fault may be more fully developed at a later time but for the present treat it as a basic event.
- o TRANSIN gate. Indicates that the tree is developed further at the occurrence of the corresponding gate symbol on the same page or on another page. The RTRANS/LTRANS shapes need not be used at the point of transfer. IRRAS 2.0 will transfer to the gate at the other location based on gate names. Therefore, the TRANSIN gate and the gate being transferred to must have the same name.
- o HOUSE event. This is a fault that can be turned on or off depending on the previous events or boundary conditions of the analysis.
- o INHIBIT gate. Output fault occurs if the single input fault occurs in the presence of an enabling condition (the enabling condition is represented by a conditional event drawn as part of the gate.)
- o NOT AND gate. The output fault occurs if at least one complement input occurs.
- o NOT OR gate. The complement of all inputs must occur for output fault to occur.
- o RTRANS/LTRANS symbols. Indicates that this portion of the tree is attached or repeated at the corresponding TRANSIN gate. These shapes are used to clarify transfers for ease of review of the trees. They do not impact the logic. Transfers are made logically using the gate and file names.
- o UDTRANS gate. An undeveloped transfer indicating that the logic may be developed at some later date but for the present it is treated as a basic event.
- o HORBOX. A horizontal box with horizontal attaching lines used to provide space for descriptive text. This shape does not impact the logic of the tree.
- o VERBOX. A horizontal box with vertical attaching lines used to provide space for descriptive text. This shape does not impact the logic of the tree.

For more information on the use of these shapes and fault tree modeling techniques, see the Fault Tree Handbook, NUREG-0492.²

4.1.3.3 Rules and Assumptions for Creating Fault Tree Logic.

A line drawing is a logical method for showing an organization structure such as a fault tree. The graphical depiction gives a quick visual reference as to which entity is a member of a family and which entities are submembers (or children) of parent entities. While the graphical representation is excellent for a visual overview of related members, it is not directly usable in programs for analyzing or processing the fault tree.

A path-solving algorithm bridges the gap from a graphical representation of family members to a form useful for input to computation programs. The path-solving module contains basic assumptions which are rules for building the fault trees. These are:

1. Parent and sibling shapes must be recognized. If two shapes are connected together, the parent shape is the shape with the adjoining line connected to the bottom, and the child is the shape with the line connected to the top. Since a BASIC EVENT (see Figure 120) always has the line connect to the top, it is always a child. Other shapes, such as AND or OR gates, connect at top and bottom. In Figure 121, A is parent to B.

Figure 121. Parent and child shape relationship.

In Figure 122, both shapes connect to the bottom. One shape will arbitrarily be chosen as the parent.

Figure 122. Arbitrary shape relationship.

2. It is not convenient and often impossible to connect all objects using a single straight line. Thus, lines which intersect are in the same ancestry path. In Figure 123, lines L1, L2, and L3 intersect. Thus, shapes B, C, D, and E are on the same level and children to A. Shape B has child F.

Figure 123. Relationships with intersecting lines.

3. There is no provision for jumping across a path line. An attempt to bridge a line, such as in Figure 124, results in D and E being unrelated.

Figure 124. Attempting to jump lines.

4.1.3.4 The Graphics Attribute Header. When building a fault tree, the current attributes are shown at the top of the screen. These attributes reflect the current settings. Figure 125 illustrates the attribute list. A description of items in the attribute list is given without reference to the commands that set the attributes.

Figure 125. The graphics attribute list.

The first item in the attribute list is the file name of the current drawing. An "*" at the first of the list indicates that a new drawing is being built. If the drawing has been saved (see SAVE and LOAD TREE commands on the FILE submenu), the file name is shown at the first of the list.

The boxes following each of the next four items, LINE, FILL, TEXT, and NAME, give the current line color, shape fill color, text description color, and shape name color, respectively. The line color is the color of the lines drawn between shapes. The shape boundary is always white.

The TEXT= indicator shows the current height for the description text. The unzoomed screen extends 120 units in the horizontal direction and 66 in the vertical direction.

The descriptive text horizontal justification is noted by C (center), R (right) or L (left). The justification is relative to the picked location for writing the text. The shape names are always center-justified.

The indicator MULTI-PICK specifies that commands can be repeated without re-picking the command. The repeated command is terminated by the right button on the mouse. The indicator SINGLE-PICK specifies that commands cannot be repeated without re-picking the command.

The SHOW= indicator specifies whether description text or shape names will be shown. The T specifies that descriptive text will be shown, and N specifies that the shape names will be shown. An NT specifies that descriptive text will not be shown, and similarly, an NN specifies shape names will not be shown.

The last item in the attribute list reflects the current line type. New lines connecting shapes will be drawn in this line style. If the drawing filename (the first item in the attribute list) is long, the line type may not be shown.

4.1.3.5 Main Graphics Menu. The main graphics menu is put up by moving the cursor to the desired location and pressing the left button on the mouse. The main graphics menu will be displayed at that location, as illustrated in Figure 116. To move the main menu, move the screen cursor to another location (outside of the menu) and press the left button. The menu will move to the new location. It can be taken down by putting the cursor on the menu and pressing the right button on the mouse. Sub-menus and commands can be picked using the cursor left button. The commands in the main graphics menu are:

- o **SHOW.** The screen is blanked and the picture is redrawn.
- o **BUILD.** The BUILD submenu is displayed. The BUILD menu contains the predefined fault tree shapes. (See Section 4.1.3.6).
- o **EDIT.** The EDIT submenu is displayed. The EDIT menu allows parts of the picture to be moved, copied, or deleted. (See Section 4.1.3.8).
- o **TEXT.** The TEXT submenu is displayed. The TEXT menu allows text, shape names, and transfer names to be entered or identified. (See Section 4.1.3.9).
- o **VIEWS.** The VIEWS submenu is displayed. The VIEWS menu allows fault tree drawings to be enlarged, reduced, or panned. A grid is also available on this menu to aid in aligning fault tree shapes. (See Section 4.1.3.11).
- o **SET UP.** The SET UP submenu is displayed. The SET UP menu allows general attributes and attributes for the shapes, shape names, and text to be set. (See Section 4.1.3.13).
- o **FILE.** The FILE submenu is displayed. The FILE menu allows pictures to be retrieved, saved, or created (See Section 4.1.3.14).
- o **EXIT.** The graphics processor is terminated. Control will be returned to the IRRAS-PC Fault Tree Graphics System menu. If the current drawing has been modified but not saved, a message is displayed at the bottom of the screen asking whether to continue with the EXIT command. Enter X to continue or any other character to abort the command. If the

EXIT command is continued, the last modifications to the current drawing will be lost.

4.1.3.6 The BUILD Submenu. The BUILD submenu contains the predefined fault tree shapes and is entered by picking BUILD on the main graphics menu using the left button on the mouse. The BUILD submenu is shown in Figure 119.

Fault tree shapes corresponding to the menu names are shown in Figure ?. A shape is picked by moving the cursor to the menu name and pressing the left button on the mouse. The right button will take down the BUILD submenu. The picked shape is shown on the screen as a ghost which can be moved to the desired location. Once located, the ghost is permanently affixed by pressing the left button a second time. The second depression of the left button terminates the command if you are in the SINGLE-PICK mode. Another shape then can be picked from the menu. If the MULTI-PICK option is in force, then the ghost remains in hand and can be placed at multiple locations using the left button. Pressing the right button drops the shape and terminates the command.

If the wrong shape is picked and detected before it is set in place, press the right button on the mouse; the ghost is dropped and another choice can be made. If a wrong shape is discovered after it has been set, it can be erased using the DELETE command in the EDIT submenu.

As each shape is picked and placed in the fault tree, it is given a default name. The default name for a gate is GATEn where n is the next available sequential number. The default name for a basic event is EVENTn. The alphabetic part can be set, using the DFLT GATE or DFLT EVENT command located in the SETUP menu, or left at the IRRAS default characters. The number is added to the characters to give a unique name. After a shape is placed in the fault tree, the shape can be renamed by using the EDIT NAME command, located in either the TEXT menu or EDIT menu, or by using the REP NAME command located in the TEXT menu. A duplicate name can be entered, but the user is questioned whether the duplicate name is valid.

BUILD commands are evoked by picking the menu command names using the left button on the mouse. The BUILD commands are:

- o Attributes. Puts up the BUILD Attributes submenu. This submenu, shown in Figure 126, allows attributes pertaining to shapes and lines to be set. These attributes can also be set in the SET UP menu.

Figure 126. The BUILD Attributes Submenu.

- o LINE. Line segments are drawn using the left button on the mouse. Each time the button is pressed, a line is drawn from the last location to the new location. If SINGLE-PICK is in force, the right button terminates the LINE command. If MULTI-PICK is in force, the right button terminates the current connected line segments. Pressing the left button starts a new set of line segments. The line command, in the MULTI-PICK mode, is terminated by pressing the right button twice. Line segments are drawn in the current line color and style.
- o AND. Draws an AND shape. Output fault occurs if all of the input faults occur.
- o OR. Draws an OR shape. Output fault occurs if at least one of the input faults occurs.
- o BEVENT. Draws a BASIC EVENT shape. The basic event shape indicates that the fault requires no further development.

- o TABLE. Draws a TABLE shape. The TABLE shape contains a tabular list of events. Up to 8 basic events can be listed. The table is initially empty. To fill in the table, use the the TABLE ENTRY command in the TEXT menu.
- o N/M OR. Draws an N/M OR shape. The N/M OR gate requires N events out of a maximum of M events to be true for the gate to be true. After the N/M OR gate is placed, a request is made for the N and M values to be entered. Using the keyboard, enter the two numbers. Alphabetic characters can be entered with the numbers. The first two numbers encountered will be the N and M values, respectively. Examples of valid N and M entries are:

2/3 2 3 2 out of 3 23

where the N value is 2 and the M value is 3. The allowed range of values for N and M is 1 through 9.

For the N/M OR gate to be valid, it must have M and only M children. Thus if M is 3, then there must be 3 gates and/or events connected to the bottom of the gate. No check is made while building the fault tree as to whether a valid number of gates and/or events have been entered. See Section 6.1 for details on the processing of an N/M OR gate.

- o UEVENT. Draws an undeveloped event shape. The undeveloped event indicates that the logic for the fault may be more fully developed at a later time but for the present it is treated as a basic event.
- o TRANSIN. Draws a TRANSIN shape. The TRANSIN gate indicates that the tree is developed further at the occurrence of another gate on the same page or on another page (another fault tree picture). IRRAS 2.0 will transfer to the gate at the other location. The TRANSIN shape name is the name of the gate to which the logic transfers. When a TRANSIN shape is initially placed in the drawing, the shape is given a default name (TRANSn). The transfer name can be specified by using the REP NAME command located in the TEXT menu. A duplicate transfer name can be entered, but the user is questioned as to whether the duplicate name is valid. If the specified transfer gate does not exist locally on the picture or on another page, the TRANSIN is simply a basic event.

To make a transfer to another page, place a TRANSIN shape on the tree in the desired location and give it a name of up to 16 characters, with the first 8 characters being unique from other fault tree names in the family. The fault tree page being transferred in must have the same name as the first 8 characters of the TRANSIN gate. The top gate on the TRANSIN

page must have the same name as the TRANSIN gate. Note that a transfer to another page must go to the top gate on that page. To transfer to another gate on the same page, place the TRANSIN on the drawing and name it the same as the gate being transferred in.

- o HOUSE. Draws a HOUSE event shape. This is a fault that can be turned on or off depending on the previous events or boundary conditions of the analysis.
- o INHIBIT. Draws an INHIBIT shape. Output fault occurs if the single input fault occurs in the presence of an enabling condition. The enabling condition is represented by a conditional event (CEVENT) drawn as part of the gate. Similar to other shapes, the associated CEVENT is given a default name. The conditional event is changed using REP CEVENT located in the TEXT menu.
- o NOT AND. Draws a NOT AND gate shape. The logic is satisfied unless all of the input faults occur.
- o NOT OR. Draws a NOT OR gate shape. The logic is satisfied if none of the input faults occur.
- o RTRANS (LTRANS). Draws a RTRANS (LTRANS) shape. The RTRANS (right) AND LTRANS (left) indicates that this portion of the tree is attached or repeated at the corresponding TRANSIN gate. These shapes are used to clarify transfers for ease of reviewing trees. They do not impact the logic. Transfers are made logically using the gate and file names.
- o UDTRANS. Draws an undeveloped transfer shape. An undeveloped transfer indicates that the logic may be developed at some later date, but for the present it is treated as a basic event.
- o HORBOX. Draws a horizontal box shape. A box with horizontal attaching lines used to provide space for descriptive text. This shape does not impact the logic of the tree.
- o VERBOX. Draws a vertical box shape. A box with vertical attaching lines used to provide space for descriptive text. This shape does not impact the logic of the tree.

4.1.3.7 BUILD Attributes Submenu. The shape and line attributes are set from a submenu of the BUILD menu. The attribute submenu is entered by picking Attributes located at the top of the BUILD menu using the left-button on the mouse. The BUILD Attribute submenu is shown in Figure 126.

Attributes are set by picking commands from the BUILD Attributes submenu using the left button on the mouse. The submenu is taken down by placing the cursor on it and pressing the right button on the mouse.

The options on the BUILD Attributes submenu are described as follows:

- o FILL COL. Sets the fill color. The fill color applies only to shapes. A pop-up color submenu appears, and the color is picked from the submenu. The current fill color is shown in the attribute list at the top of the screen.
- o LINE COL. Sets the line color. The line color applies to lines drawn with the LINE command. A pop-up color submenu appears, and the color is picked from the submenu. Lines around the shapes are always white. The line color is shown in the attribute list at the top of the screen.
- o LINE TYPE. Sets the line type. The line type applies to lines drawn with the LINE command. The pop-up line type submenu appears, and the line type is chosen. All line types are treated as solid lines by the logic analysis portions of IRRAS 2.0.
- o TEXT COL. Sets the description text color. A pop-up color submenu appears, and the color is picked from the submenu. The current description text color is shown in the attribute list at the top of the screen.
- o TEXT SIZE. Sets the descriptive text size. The height is entered following the prompt at the bottom of the screen. The default screen size is 120 units in the horizontal direction and 66 units in the vertical direction. The current text size is shown in the attribute list at the top of the screen.
- o TEXT JUSTIFY. This option sets the description text horizontal justification. A pop-up justification menu appears, and either LEFT, CENTER, or RIGHT is picked from the submenu. The current description text justification (R for right, C for center, and L for left) is shown in the attribute list at the top of the screen.
- o NAME COL. Sets the shape name color. A pop-up color submenu appears, and the color is picked from the submenu. The current name color is not shown in the attribute list at the top of the screen.
- o NAME SIZE. Sets the shape name size. The height is entered following the prompt at the bottom of the screen. The

default screen size is 120 units in the horizontal direction and 66 units in the vertical direction.

4.1.3.8 The EDIT Submenu. Using commands on the EDIT submenu, parts of a diagram can be moved, copied, or deleted and events in a TABLE of events can be modified. The EDIT submenu is entered by picking EDIT on the main graphics menu using the left button on the mouse. The EDIT submenu is shown in Figure 127. All selections on the EDIT submenu are made using the left button on the mouse. The submenu is taken down using the right button.

Figure 127. The EDIT Submenu.

The options on the EDIT submenu are described as follows:

- o COPY and COPY Text. The COPY command copies all objects in an identified part of the screen. The COPY Text command copies only text in an identified part of the screen. To copy, place the screen cursor at one corner of a box encompassing the region to be copied and press the left button. Move to the other corner of the box. A rubber band box will follow the cursor as it is moved. Once the box is properly located, press the left button. All items (or all text if COPY Text command) contained completely within the box will be copied. If the box is not located properly, press the right button to cancel the COPY command and start again. Shapes, lines, or lines of text only partially inside the box will not be copied. Pick a reference point for copying and press the left button. Move the cursor to the location to place the copy and again press the right button. All parts (or all text if COPY Text command) of the chosen region will be copied to the new location with the reference point laying on the new point. The copied gates and events

will be given new default names. If the MULTI-PICK option is enabled, the COPY command is active until the right button on the mouse is pressed.

- o DUPLICATE. To copy a section of a drawing and have the gates and events retain their original names, the DUPLICATE command is used. The procedure for duplicating sections of a drawing is the same as that for the COPY command. If the MULTI-PICK option is enabled, the DUPLICATE command is active until the right button on the mouse is pressed.
- o DELETE. To delete sections of the drawing, select the DELETE command. Place the screen cursor at one corner of a box encompassing the region to be deleted and press the left button. Move to the opposite corner of the box. A rubber band box will follow the cursor as it is moved. Once the box is properly located, press the left button on the mouse. All items completely within the box will be deleted. Lines, shapes, and text only partially within the box will not be affected. If the MULTI-PICK option is enabled, the DELETE command is active until the right button on the mouse is pressed.
- o MOVE and MOVE Text. To move a section of the drawing, pick MOVE or MOVE Text on the EDIT submenu. The MOVE command moves all objects, while the MOVE Text command moves only text. Place the screen cursor at one corner of a box encompassing the region to be moved. Press the left button. Move to the other corner of the box. A rubber band box will follow the cursor as it is moved. Once the box is properly located, press the left button on the mouse. All items completely within the box are marked to be moved. Shapes, lines, or text only partially inside the box will not be moved. Using the screen cursor, pick a reference point for moving. Move the cursor to the new location to place the region and again press the left button. All parts of the chosen region will be moved to the new location, with the reference point laying on the new point. If the MULTI-PICK option is enabled, the MOVE command is active until the right button on the mouse is pressed.

4.1.3.9 The TEXT Submenu. Text and shape names are entered using the TEXT submenu. This menu is entered by picking TEXT on the main graphics menu using the left button on the mouse. The TEXT submenu is shown in Figure 128.

Figure 128. The TEXT submenu.

The options on this menu are described as follows:

- o Attributes. Puts up the TEXT Attributes submenu. This menu, shown in Figure 129, allows attributes pertaining to text to be set. These attributes can also be set in the SET UP menu.

Figure 129. The TEXT Attributes submenu.

- WRITE TEXT. The WRITE TEXT command initiates entry of descriptive text. It is selected using the left-button on the mouse. Move the cursor to the location for the bottom edge of the first line of text and press the left-button. The text will be left-, center-, or right-justified depending on the horizontal justification setting (shown in the attribute list at the top of the screen). Descriptive text can be placed anywhere on the screen. The text is entered after the prompt at the bottom of the screen. Multiple stacked lines of text can be entered. Terminate each line of text with <Enter>. Maximum number of characters allowed per line of text is 80. To terminate entry of text, press the <Esc> key or press the <Enter> key as the only entry. The first descriptive text line is left-justified, centered, or right-justified relative to the specified location point, with additional lines placed under the previous line. If the MULTI-PICK option is enabled, the WRITE TEXT command is active until the right button on the mouse is pressed.
- REP NAME. As each shape is entered, it is given a default name. The shape name can be changed (replaced) using the REP NAME command. The assignment of a new name to a shape is initiated by picking REP NAME on the TEXT submenu and then picking the shape to be named. The replacement name is entered after the prompt at the bottom of the screen.

The name of a basic event is limited to 16 characters. A complemented event (a not event) is indicated by preceding the name with a slash (/). If the name is a duplicate name, a message is issued questioning whether the duplicate name is valid. If it is, answer Y. If it is not, answer N. If the MULTI-PICK option is enabled, the command is active until the right button on the mouse is pressed.

- o SHOW NAME. At times the names are too small to read on the screen. To display the name of a shape, pick SHOW NAME and then pick the shape. The name of the shape is displayed at the bottom of the screen. If the MULTI-PICK option is enabled, the SHOW NAME command is active until the right button on the mouse is pressed.
- o FIND NAME. To find all shapes with a given name, the FIND NAME command is picked. The characters used to locate the name are entered after the prompt at the bottom of the screen. A box is drawn around each shape whose name begins with the specified characters. For example, if the given characters are "Ga", then shapes with names "Ga", "Gat", "Gate", etc. will be located. If the MULTI-PICK option is enabled, the FIND NAME command is active until the right button on the mouse is pressed.
- o CLEAR FIND. To remove the box around the located shapes (from FIND NAME command), pick CLEAR FIND.
- o TABLE ENTRY. The loading of events into a TABLE is initiated by picking TABLE ENTRY in the TEXT menu and then picking the TABLE shape to receive the events. The event names are entered after the prompt at the bottom of the screen. If the name is a duplicate name, a message is issued questioning whether the duplicate name is valid. If it is, answer Y. If it is not, answer N and the name is ignored. To terminate entry of events, enter <Esc> or <Enter>. The color and height of the TABLE event names are the same as those for the shape name. Events also can be entered using the EDIT NAME command. If the MULTI-PICK option is enabled, the command is active until the right-button on the mouse is pressed.
- o EDIT TABLE. The EDIT TABLE command allows shape names or events in a TABLE to be edited. To change event names, pick EDIT TABLE from the TEXT menu, and then pick the shape to be modified. A menu showing the shape name or events, if a TABLE, is displayed as illustrated in Figure 130. The "@" character signifies an empty event slot.

Figure 130. The EDIT TABLE menu list.

To delete an event, pick DELETE and then the name to delete. To delete another event, simply pick the next name. The delete process is terminated by picking REPLACE. Picking outside the EDIT TABLE menu terminates the EDIT TABLE command.

To replace an event, pick REPLACE to initiate the command. Then pick the entry to be changed. Type the replacement entry after the prompt at the bottom of the screen. A new basic event is added by picking a "@" (an empty slot in the list) and entering the new name. To replace another event, pick the next name to be replaced. The replace process is terminated by picking DELETE. Picking outside the EDIT TABLE menu terminates the EDIT TABLE command.

- REP CEVENT. The INHIBIT gate's associated conditional event name can be changed by picking REP CEVENT in the TEXT menu and then picking the INHIBIT gate. The conditional event name is entered after the prompt at the bottom of the screen. If the name is a duplicate name, a message is issued questioning whether the duplicate name is valid. If it is, answer Y. If it is not, answer N. If the MULTI-PICK option is enabled, the command is active until the right button on the mouse is pressed.
- SHOW CEVENT. To show the INHIBIT gates's associated conditional event name, pick SHOW CEVENT and then pick the INHIBIT shape. The name is displayed at the bottom of the

screen. If the MULTI-PICK option is enabled, the command is active until the right button on the mouse is pressed.

4.1.3.10 The TEXT Attributes Submenu. The text attributes are set from a submenu of the TEXT menu. The attribute submenu is entered by picking Attributes located at the top of the TEXT menu using the left button on the mouse. The TEXT Attribute submenu is shown in Figure 129.

Attributes are set by picking commands from the TEXT Attributes menu using the left button on the mouse. The submenu is taken down by placing the cursor on the menu and pressing the right button on the mouse.

The options on this submenu are described as follows:

- o TEXT COL. This option sets the description text color. A pop-up color submenu appears, and the color is picked from the submenu. The current description text color is shown in the attribute list at the top of the screen.
- o TEXT SIZE. This option sets the descriptive text size. The height is entered following the prompt at the bottom of the screen. The default screen size is 120 units in the horizontal direction and 66 units in the vertical direction. Both the current text height and color are shown in the attribute list at the top of the screen.
- o TEXT JUSTIFY. This option sets the description text horizontal justification. A pop-up justification menu appears and either LEFT, CENTER, or RIGHT is picked from the submenu. The current description text justification (R for right, C for center, and L for left) is shown in the attribute list at the top of the screen.
- o NAME COL. This option sets the shape name color. A pop-up color submenu appears, and the color is picked from the submenu. The current name color is not shown in the attribute list at the top of the screen.
- o NAME SIZE. This option sets the name size. The height is entered following the prompt at the bottom of the screen. The default screen size is 120 units in the horizontal direction and 66 units in the vertical direction.
- o DFLT GATE and DFLT EVENT. Each new shape is given a unique name composed of a default name and a unique number. The initial default names for gates and events are GATE and EVENT, respectively. To change the default gate name, pick DFLT GATE from the TEXT Attributes submenu and enter the name after the prompt at the bottom of the screen. To change the default event name, pick DFLT EVENT and enter the

name after the prompt at the bottom of the screen. Gate and event names can be any continuous string of up to 16 characters. Spaces are not allowed and will cause analysis errors if used. Certain characters (*, /, ?, \$) should be avoided, since they have specific meanings in IRRAS 2.0 or DOS.

4.1.3.11 The VIEWS Submenu. The VIEWS submenu allows fault tree drawings to be enlarged, reduced, or panned. The VIEWS submenu is entered by picking VIEWS on the main graphics menu using the left button on the mouse. The VIEWS menu is shown in Figure 131.

Figure 131. The VIEWS submenu.

The options on this submenu are described below.

- o ZOOMIN. The picture can be zoomed in to show finer detail. Select ZOOMIN and place the screen cursor at one corner of a box encompassing the region to be viewed. Press the left button. Move to the other corner of the box. A rubber band box will follow the cursor as it is moved. Once the box is properly located, press the left button on the mouse. The screen will be blanked, and the section of picture boxed will be redrawn giving higher resolution. Plus marks reference the lower-left and upper-right default locations of the screen. If the picture is zoomed in or out, one or both of the marks may be out of view. If the MULTI-PICK option is enabled, the ZOOMIN command is active until the right button on the mouse is pressed.

- o ZOOMOUT. The picture can be zoomed out to show coarser detail and a larger part of the picture. After selecting ZOOMOUT, the screen will be blanked and the picture space (window) will double in size. Shapes on the screen will be shown half their original size. Plus marks reference the lower-left and upper-right default locations of the screen. If the picture is zoomed in or out, one or both of the marks may be out of view. If the SINGLE-PICK option is enabled, the attribute list at the top of the screen is shown and the command is terminated. If the MULTI-PICK option is enabled, the picture can be repeatedly zoomed out by pressing the left button. To deactivate the ZOOMOUT command, press the right button on the mouse.
- o ZOOMRES. The picture can be reset to its original detail using the ZOOMRES command. After selecting ZOOMRES, the screen will be blanked and the picture redrawn at its original size. The original size is the resolution active when the fault tree drawing was last saved. If the drawing is new, then the ZOOMRES returns to the default size.
- o MOVE RIGHT. When the picture is zoomed in, only a portion of the picture is visible on the screen. The MOVE RIGHT command lets the user scroll to the right without changing the zoom setting. To pan right, pick the MOVE RIGHT command using the left button on the mouse. The screen will be blanked, and the picture space will be moved to the right. If the SINGLE-PICK option is enabled, the attribute list at the top of the screen is shown and the command is terminated. If the MULTI-PICK option is enabled, the picture can be repeatedly panned right by pressing the left button. To deactivate the pan command, press the right button on the mouse.
- o MOVE LEFT. When the picture is zoomed in, only a portion of the picture is visible on the screen. The MOVE LEFT command lets the user scroll to the left without changing the zoom setting. The screen will be blanked, and the picture space will be moved to the left. If the SINGLE-PICK option is enabled, the attribute list at the top of the screen is shown and the command is terminated. If the MULTI-PICK option is enabled, the picture can be repeatedly panned left by pressing the left button. To deactivate the pan command, press the right button on the mouse.
- o MOVE DOWN. When the picture is zoomed in, only a portion of the picture is visible on the screen. The MOVE DOWN command lets the user scroll down without changing the zoom setting. To pan down, pick the MOVE DOWN command using the left button on the mouse. The screen will be blanked, and the picture space will be moved down. If the SINGLE-PICK option is enabled, the attribute list at the top of the

screen is shown and the command is terminated. If the MULTI-PICK option is enabled, the picture can be repeatedly panned down by pressing the left button. To deactivate the pan command, press the right button on the mouse.

- MOVE UP. When the picture is zoomed in, only a portion of the picture is visible on the screen. The MOVE UP command lets the user scroll up without changing the zoom setting. To pan up, pick the MOVE UP command using the left button on the mouse. The screen will be blanked, and the picture space will be moved up. If the SINGLE-PICK option is enabled, the attribute list at the top of the screen is shown and the command is terminated. If the MULTI-PICK option is enabled, the picture can be repeatedly panned up by pressing the left button. To deactivate the pan command, press the right button on the mouse.
- NO NAME. The NO NAME command is a toggle for turning on or off the showing of gate and event names. Picking NO NAME suppresses names from being drawn. The suppression of names expedites the drawing of a picture and may be useful when outputting the fault tree to a pen plotter or laser printer. To show the names again, the NO NAME command is picked again. The attribute list at the top of the screen reflects whether names will be drawn. NN indicates no names will be drawn. N indicates names will be drawn.
- NO TEXT. The NO TEXT command is a toggle for turning on or off the showing of text. Picking NO TEXT suppresses text from being drawn. The suppression of text expedites the drawing of a picture when large amounts of text exist. To show the text again, the NO TEXT command is picked again. The attribute list at the top of the screen reflects whether text will be drawn. NT indicates no text will be drawn. T indicates text will be drawn.
- GRID. The GRID command toggles on or off grid lines which may be used to better align shapes. When the GRID command is activated, the picture is redrawn with grid lines. When the GRID command is deactivated, the picture is not redrawn and the grid lines remain until the SHOW command is evoked.

4.1.3.12 The SETUP Submenu. The SETUP submenu contains commands for setting up the default values for the drawing of fault trees. All selections on the SETUP submenu are made using the left button on the mouse. The SETUP submenu is shown in Figure 132.

Figure 132. The SETUP submenu.

The options on this submenu are described below.

- o FILL COL. Sets the fill color. The fill color applies only to shapes. A pop-up color submenu appears, and the color is picked from the submenu. The current fill color is shown in the attribute list at the top of the screen.
- o LINE COL. Sets the line color. The line color applies to lines drawn with the LINE command. A pop-up color submenu appears, and the color is picked from the submenu. Lines around the shapes are always white. The line color is shown in the attribute list at the top of the screen.
- o LINE TYPE. Sets the line type. The line type applies to lines drawn with the LINE command. The pop-up line type submenu appears, and the line type is chosen. All line types are treated as solid lines by the logic analysis portions of IRRAS 2.0.
- o TEXT COL. Sets the description text color. A pop-up color submenu appears, and the color is picked from the submenu.

The current description text color is shown in the attribute list at the top of the screen.

- o TEXT SIZE. Sets the descriptive text size. The height is entered following the prompt at the bottom of the screen. The default screen size is 120 units in the horizontal direction and 66 units in the vertical direction. The current text size is shown in the attribute list at the top of the screen.
- o TEXT JUSTIFY. This option sets the description text horizontal justification. A pop-up justification menu appears, and either LEFT, CENTER, or RIGHT is picked from the submenu. The current description text justification (R for right, C for center, and L for left) is shown in the attribute list at the top of the screen.
- o NAME COL. Sets the shape name color. A pop-up color submenu appears, and the color is picked from the submenu. The current name color is not shown in the attribute list at the top of the screen.
- o NAME SIZE. Sets the shape name size. The height is entered following the prompt at the bottom of the screen. The default screen size is 120 units in the horizontal direction and 66 units in the vertical direction.
- o DFLT GATE. Sets the default gate name. Each new shape is given a name composed of a default name and a unique number. The initial default name for a gate is GATE. To change the default gate name, pick DFLT GATE and enter the desired default after the prompt at the bottom of the screen.
- o DFLT EVENT. Sets the default event name. Each new event is given a name composed of a default name and a unique number. The initial default name for an event is EVENT. To change the default event name, pick DFLT EVENT and enter the desired default name after the prompt at the bottom of the screen.
- o MULT PICK. Sets the mode of operation of the commands. MULT PICK is a toggle between the single-pick mode and the multi-pick mode. The multi-pick mode keeps a command in effect until cancelled by use of the right button on the mouse. The single-pick mode allows only one execution of a command and then gives back the cursor for selection of another command. The attribute header at the top of the screen indicates which mode is in effect.

4.1.3.13 The FILE Submenu. The FILE submenu contains functions for storing and retrieving fault tree drawings, transferring (paging) between related fault tree drawings, and plotting the current picture. All selections on the FILE submenu are made using the right button on the mouse. The FILE submenu is entered by picking FILE on the main graphics menu using the left button on the mouse. This submenu is shown in Figure 133.

Figure 133. The FILE submenu.

The options on this submenu are described below.

- o LOAD TREE. To load an existing drawing, pick the LOAD TREE command. Enter the drawing file name after the prompt at the bottom of the screen. If a file extension is not given as part of the file name, the file extension ".DLS" is added to the file name by IRRAS 2.0. When trying to load another drawing when the current drawing has been modified, but not saved, a message is displayed at the bottom of the screen asking whether to continue with the LOAD TREE command. Enter L to load the drawing without saving the existing drawing or any other character to abort the LOAD TREE command. If the LOAD TREE command is continued, the last modifications to the current drawing will be lost. The file name of the last drawing loaded is shown in the attribute list at the top of the screen.

- SAVE TREE. To store a new drawing, pick the SAVE TREE command. Enter the new drawing file name after the prompt at the bottom of the screen. The file extension .DLS is added to the file name if an extension is not given. If the drawing already exists, an error message is displayed at the bottom of the screen

File already exists. Do you wish to replace? <Y/N>

If Y is chosen, then the drawing replaces the old version on the file. If N is chosen, then the file is not written to the disk and any changes since the last time the drawing was saved are gone. The fault tree family structure is calculated on each saving of the file.

- NEW TREE. A new tree can be started by evoking the NEW TREE command. The current drawing is flushed, the screen is cleared, and an "*" is placed in the file position at the top of the screen in the attribute list. If the current drawing has been modified but not saved and the NEW TREE command is evoked, a message is displayed at the bottom of the screen asking whether to continue with the NEW TREE command. Enter G to continue (existing file destroyed) or any other character to abort the NEW TREE command. If the NEW TREE command is continued, the last modifications to the current drawing will be lost.
- LIST TREES. To list the fault tree drawings (files) in the current directory, pick LIST TREES from the menu. If the list is too large to fit totally in the pop-up menu, pick <Page Up> or <Page Down> to move up and down in the list. To load a drawing from the list, pick the drawing name. To exit the LIST TREES command, pick outside of the list of fault tree names.
- TRANSFER. The TRANSIN gate indicates that the tree is developed further at the occurrence of another gate on the same page or on another page (another fault tree picture). The TRANSFER command either 1) identifies the transfer gate if the gate is on the same page or 2) transfers to related drawing specified by the TRANSIN shape.

To transfer to a related gate or drawing, pick TRANSFER and then pick the TRANSIN shape specifying the related gate or drawing. The current drawing is searched first for the transfer name. All shapes with the transfer name are boxed. If no shapes are found in the current drawing, it is assumed that the transfer is to another page (file).

Fault tree drawings can be reviewed one page at a time by using the TRANSFER command. If the TRANSIN shape contains a transfer file name (see TEXT submenu), then the current drawing is removed and the specified drawing (file) is displayed. If the TRANSIN shape does not have a transfer file name, an error message is displayed at the bottom of the screen. If the current drawing has been modified but not replaced, a message is displayed at the bottom of the screen asking whether to continue with the TRANSFER command. Enter T to continue or any other character to abort the command. If the TRANSFER command is continued, the last modifications to the current drawing will be lost. The file name of the current drawing is shown in the attribute list at the top of the screen.

- o TRAN BACK. To transfer back to an earlier page of the fault tree drawing, pick TRAN BACK. A history of previous pages (transfer files) is displayed. From the list, pick the desired page. Earlier transfer pages are retained in the transfer list while pages retrieved later than the one chosen are discarded. If a LOAD TREE command is evoked, the transfer history list is flushed. If the current drawing has been modified but not replaced, a message is displayed at the bottom of the screen asking whether to continue with the TRAN BACK command. Enter T to continue or any other character to abort the command. If the TRAN BACK command is continued, the last modifications to the current drawing will be lost.
- o PLOT EPS. To draw the fault tree on an Epson FX printer/plotter, pick PLOT EPS. The picture will be redrawn in monochrome, and the screen will be dumped through the parallel port to the printer. After completion, the picture is redrawn in color.
- o PLOT HP. To draw the fault tree on a Hewlett Packard 7400 series pen plotter, pick PLOT HP. The picture will be redrawn on the screen and written to a file but not directly drawn to the pen plotter. The intermediate plot file has the same name as the current file with extension ".HPP". The actual drawing of the picture on the pen plotter is relegated to the Plot Tree (plotter) command on the IRRAS-PC Fault Tree Graphics System menu. This allows multiple drawings to be quickly produced (placed in files) and be drawn later on the pen plotter in a batch mode. Secondly, pen plotter drawings can be made on one machine and then transferred (move the *.HPP files) to another machine for drawing.
- o EXIT. To terminate the Build Graphics Trees option, pick EXIT. Control will be returned to the IRRAS-PC Fault Tree Graphics System menu. If the current drawing has been modified but not replaced, a message is displayed at the

bottom of the screen asking whether to continue with the EXIT command. Enter X to continue or any other character to abort the command. If the EXIT command is continued, the last modifications to the current drawing will be lost.

4.2 Performing Sensitivity Analyses

The IRRAS 2.0 data base is structured to allow the analyst to make temporary changes to the models for the purposes of performing sensitivity studies. Sensitivities fall into two basic categories: those that impact the logic of the models and those that impact the component and reliability data used to quantify the logic models.

4.2.1 Sensitivities Impacting the Reliability Data Base

Sensitivity studies that only impact the data used to quantify the models are the easiest to handle in IRRAS 2.0. The IRRAS 2.0 data base has space for two sets of reliability data for each basic event. The base case data is entered through the MODIFY Data Base module. This is done by one of two methods.

4.2.1.1 Establishing a Base Case Data Base. The first method is the entering of each basic event data set using the Modify Event Data option on the Edit Basic Event Data menu. This is done by selecting MODIFY Data Base from the IRRAS 2.0 main menu. This brings up the Edit Data Base Records menu shown in Figure 134

Edit Data Base Records

Option |E| Exit / Family / Basic Events
 1 - Fault Trees / 2 - Event Trees & Sequences

DEMO FAMILY Demonstration sample family

Note: Select type of data to be edited, move cursor to desired family, and press <Enter>.

Use <PgUp> or <PgDn> to display more Families.

Figure 134. Edit Data Base Records menu (option B).

On this menu, select B - Basic event, highlight the desired family, and press <Enter>. This will bring up the Edit Basic Event Data menu, as shown in Figure 135.

Edit Basic Event Data

Currently Selected Family —> DEMO FAMILY

Option | | Exit / Modify Event Data <= choose an Event
Update / Remove All unused events <= other options

C-CV-A	CCS Train A pump discharge check valve
C-CV-B	CCS Train B pump discharge check valve
C-MOV-1	CCS suction isolation valve
C-MOV-A	CCS Train A pump discharge isolation valve
C-MOV-B	CCS Train B pump discharge isolation valve
C-PUMP-A	CCS Train A motor-driven pump
C-PUMP-B	CCS Train B motor-driven pump
DG-A	Emergency diesel generator A
DG-B	Emergency diesel generator B
E-CV-A	ECS Train A pump discharge check valve
E-CV-B	ECS Train B pump discharge check valve
E-MOV-1	ECS suction isolation valve

NOTE: The Update option makes the current reliability and uncertainty values the new base case. Use <PgUp> or <PgDn> to display more Events.

Figure 135. Edit Basic Event menu.

From this menu, select M for Modify Event Data, highlight the basic event to be changed, and press <Enter>. IRRAS 2.0 will put up the Edit Event menu of Figure 136.

Edit Event

Currently Selected Family —> DEMO FAMILY

Option |M| Exit / Add / Modify / Delete / Next / Previous

Event-Attributes

Event Names	Fail Mode	Component Type	System Type	Component ID
<P> C-CV-A	A1	CV	CCS	C-CV-A
<A> C-CV-A				

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Location	FZ1															
Description	CCS Trian A pump discharge check valve															

Uncertainty-Data

Distribution Type —> L
(press Esc for list of types)

Value —> 3.000E+000

Correlation Class-> 1

Failure-Data

Calculation Type —> 1
(press Esc for list of types)

Prob —> 1.000E-004

Lamda —> +0.000E+000

Tau —> +0.000E+000

Calculated Prob-> 1.000E-004

Figure 136. Edit Event menu.

On this menu, the user can classify the event according to failure mode, component type, system type, component ID, location, and susceptibilities. Each of these fields is available for the user to input his own code. It is left to the user to keep track of the codes used in each field. A description can also be entered. Below these fields are the failure data and uncertainty data fields where the base case data can be entered. The cursor is moved around the menu with the arrow keys; and when all fields are filled in as desired, the <Enter> key is pressed. This sends the data to the data base and causes IRRAS 2.0 to display the

Record Modified

message at the bottom of the screen. Move the cursor up to the option field and select E for Exit and press <Enter> to return to the Edit Basic Event Data menu. This process is repeated for each basic event.

The second method of creating a base case data base is to take the current case data base developed as described in Section 3.4 and copy it. This is done by entering the MODIFY Data Base module, selecting the B - Basic event option, highlighting the desired family, and pressing <Enter>. On the Edit Basic Event Data menu, select the U - Update option. IRRAS 2.0 displays the message

Updating base case event values with current values
and then displays

Base event values have been updated

This message indicates that the base case data base has been changed to be the same as the current case data base.

NOTE: Updating causes the base case data to be overwritten with the current case data with no method of retrieval. Handle with care!

4.2.1.2 Creating a Sensitivity Analysis Data Base. Once a base case data base has been established and the base case analyses have been performed (as described in Section 3), the sensitivity analysis data base can be created and used to re-evaluate the models.

The sensitivity analysis data base is created by making changes to the affected portions of the base case. By using the Reset to Base Case option on the Set Event Data menu (Figure 31), the current case data base is assured to be the same as the base case data base.

NOTE: Using the Reset to Base Case option without establishing a base case data base will result in the loss of all data previously input to the current data base through the Modify Event Data option in the FAULT Tree Analysis and SEQUENCE Analysis modules.

Next, the appropriate changes are made to the affected basic events and initiating events by using the Modify Event Data option on the IRRAS-PC Fault Tree Analysis System menu (Figure 30) or the IRRAS-PC Sequence Analysis System menu (Figure 75). Section 3.4 describes how the Probability option on the Set Event Data menu (Figure 137) is used. This option is good for making relatively few changes to the data base. For more extensive changes, the Class option on the Set Event Data menu may prove more useful.

S e t E v e n t D a t a

Option |E| Exit / Probability / Fault Tree list / Seq. list
<← choose Event

Class / Reset to Base Case
<← other options

N a m e	D e s c r i p t i o n
C-CV-A	CCS Train A pump discharge check valve
C-CV-B	CCS Train B pump discharge check valve
C-MOV-1	CCS suction isolation valve
C-MOV-A	CCS Train A pump discharge isolation valve
C-MOV-B	CCS Train B pump discharge isolation valve
C-PUMP-A	CCS Train A motor-driven pump
C-PUMP-B	CCS Train B motor-driven pump
DG-A	Emergency diesel generator A
DG-B	Emergency diesel generator B
E-CV-A	ECS Train A pump discharge check valve
E-CV-B	ECS Train B pump discharge check valve
E-MOV-1	ECS suction isolation valve

Note: Use <PgUp> or <PgDn> to display more Events.

Figure 137. Set Event Data menu with descriptions.

The Class option is selected by typing C for Class changes and pressing <Enter>. This brings up the Event Class Changes menu shown in Figure 138.

Event Class Changes

Currently Selected Family → DEMO FAMILY

Event-Attributes	
Event Name	Fail Mode
Component Type	System Type
Component ID	
	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Location	Class Attributes

Note: Enter the attributes of the events to be changed. Fields which are left blank are not used. The ? may be used for positional "don't care" characters. The * may be used for non-positional variable length "don't care" strings.
e.g. AB?CD* matches ABXCD through ABXCDXXXXXXXXXX

Uncertainty-Data

Distribution Type →
(press Esc for list of types)

Value → -----E----

Correlation Class → ----

Failure-Data

Calculation Type →
(press Esc for list of types)

Prob → -----E----

Lamda → -----E----

Tau → -----E----

Figure 138. Event Class Changes menu.

This menu allows the user to define a specific group or class of events to which the desired changes apply. The definition of the group may be based on any of the fields in the Event Attributes area of the menu. A blank field indicates that the attribute is not part of the group definition.

As an example, let's say that we wanted to change the failure probability of all check valves in the DEMO FAMILY. If we had classified the basic events according to component type, we could fill in the Component Type field with the code for check valve and then make the appropriate changes to the failure and uncertainty data. In this example, a naming scheme for the basic events that contained the component type was used instead.

So, to change the failure data for all check valves in the data base (C-CV-A, C-CV-B, E-CV-A, E-CV-B), we moved the cursor to the Event Name field with the arrow keys or tab key. In this field, we entered the desired sort criteria. In our naming scheme, the first character (C) indicated the system type and the second character was a hyphen. The third and fourth characters (CV) indicated the component type. These were followed by a hyphen and a specific component designator (A or B).

We used wild card characters and the component type code to specify all check valves in the data base. A question mark (?)

was used to indicate a position in a name that did not matter to the sort. The code for check valve, CV, was entered in the positions that this code appears in the basic event names. An asterisk (*) was used to indicate that whatever followed the component type code did not matter to the sort. Thus, our entry in the Event Name field was

??CV*

Next, we used the arrow keys or tab key to move the highlight to the failure data area and the Prob field. Here we entered the new failure probability for check valves, 3.5E-4. It might be noted at this time we could use calculation type 8 or 9 to do a sensitivity analysis. These calculation types allow us to modify current probability by a certain percentage or by a fixed amount. Remember the calculation type will default to 1. At this point, the menu looked like Figure 139.

Event Class Changes

Currently Selected Family —> DEMO FAMILY

Event Name		Fail Mode	Event-Attributes																	
			Component Type	System Type															Component ID	
??CV*				1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
Location	Class Attributes																			

Note: Enter the attributes of the events to be changed. Fields which are left blank are not used. The ? may be used for positional "don't care" characters. The * may be used for non-positional variable length "don't care" strings.
e.g. AB?CD* matches ABXCD through ABXCDXXXXXXXXXX

Uncertainty-Data

Distribution Type —>
(press Esc for list of types)

Value —> -----E----

Correlation Class-> ----

Failure-Data

Calculation Type —>
(press Esc for list of types)

Prob —> 3.500E-004

Lamda —> -----E----

Tau —> -----E----

Figure 139. Event Class Changes menu for check valve data change.

We pressed the <Enter> key to send the changes to the data base. IRRAS 2.0 indicated that the changes were made by putting up the message

4 basic events effected by the specified changes

and returned to the Set Event Data menu. From here, we exited and proceeded to re-analyze the models with the new data.

4.2.1.3 Completing the Sensitivity Analysis. After all the fault trees and event tree accident sequences are re-analyzed (see Sections 3.5 and 3.7) for the sensitivity study, the results are obtained and documented through the REPORT on Data Base module (see Section 3.8). Reports on the sensitivity study results can be sent to files on the hard disk or directly to the printer. Be sure to clearly indicate the sensitivity study in the titles of the reports.

NOTE: Be sure to complete ALL fault tree re-analyses before obtaining sensitivity reports. Failure to do so may cause erroneous results. Cut set values are not stored in the data base, but are recalculated for each report requested. By changing an event probability and obtaining a cut set report without requantifying the fault tree, you will create a disparity between the cut set values and the minimum cut set upper bound displayed in the report.

4.2.2 Sensitivities Impacting the Logic Models

Sensitivity studies that impact the logic models involve such things as changing the configuration of a system, changing the scope of the analysis (adding common-cause failures, for example), and changing dependencies among systems. These types of changes require the redefinition of the logic models and requantification.

System models are changed without destroying the base case by putting the base case models under two names in the data base. Section 3.3.5.7 explains how this is done. One name indicates the base case and the other name indicates the sensitivity study case. Modifications are made to the sensitivity study case to properly model the sensitivity study issue.

If the sensitivity study requires new accident sequences, they can be created under another event tree name (see Section 3.7), thereby preserving the base case. The processing of the new models is performed in exactly the same manner as the base case. Reports on the sensitivity study model results should be given titles that clearly indicate the sensitivity study.

4.3 Adding Recovery Actions

To add recovery actions the user must go into the Cut Set Editor. See Section 3.5.4 for further information.

4.4 Converting Studies Done on SETS to IRRAS

Because of the versatility of IRRAS 2.0 (not to mention its low cost), it is desirable to convert previously completed studies to the IRRAS 2.0 system. IRRAS 2.0 has the ability to convert SETS³ input files to the IRRAS format.

4.4.1 Converting SETS Input Files

Figure 140 is a listing of an input file used to perform fault tree analysis using SETS. To convert this file to the IRRAS format, it must be put in the current directory under a name ending with the suffix .SET. Since this input file is for the containment cooling system used in the example in Section 3, this file was named CCS1.SET.

NOTE: Within the IRRAS system, the .SET suffix indicates a file in the SETS input format. Only files with this suffix will be converted to the IRRAS format, and all files converted from IRRAS 2.0 to the SETS format will be given this suffix.

```
OG$ CCS-TOP.          IN$ DG-B, CCS-TRAINS, dummyGate1.
AG$ CCS-TRAINS.        IN$ CCS-TRAIN-A, CCS-TRAIN-B.
                        OUT$ CCS-TOP.
OG$ dummyGate1.        IN$ CCS-MOV-1, TANK.
                        OUT$ CCS-TOP.
OG$ CCS-TRAIN-A.       IN$ C-CV-A, C-MOV-A, DG-A, C-PUMP-A.
                        OUT$ CCS-TRAINS.
OG$ CCS-TRAIN-B.       IN$ DG-B, dummyGate2.
                        OUT$ CCS-TRAINS.
OG$ dummyGate2.        IN$ C-MOV-B, C-CV-B, C-PUMP-B.
                        OUT$ CCS-TRAIN-B.
BE$ C-CV-A. OUT$ CCS-TRAIN-A.
BE$ C-CV-B. OUT$ dummyGate2.
BE$ C-MOV-A. OUT$ CCS-TRAIN-A.
BE$ C-MOV-B. OUT$ dummyGate2.
BE$ C-PUMP-A. OUT$ CCS-TRAIN-A.
BE$ C-PUMP-B. OUT$ dummyGate2.
BE$ CCS-MOV-1. OUT$ dummyGate1.
BE$ DG-A. OUT$ CCS-TRAIN-A.
BE$ DG-B. OUT$ CCS-TOP, CCS-TRAIN-B.
BE$ TANK. OUT$ dummyGate1.
```

Figure 140. SETS input file, CCS1.SET.

With this file in the current directory, IRRAS 2.0 was entered and the UTILITY Options module from the IRRAS 2.0 Main menu was selected. This brought up the IRRAS-PC Utilities menu shown in Figure 141.

Figure 141. IRRAS-PC Utilities menu (option I).

On this menu, option I, IRRAS/SETS Interface, was selected. This option allows file conversion from IRRAS to SETS as well as from SETS to IRRAS. The IRRAS to SETS Interface menu is displayed in Figure 142.

Figure 142. IRRAS-PC IRRAS-to-SETS Interface menu (option X).

On this menu, the current family was checked to be correct. If it was not, it should be changed as described in Section 3.3.1. With the correct family selected, option X - SETS to IRRAS was chosen. An additional submenu was displayed showing the various conversion options. This menu is shown in Figure 143.

Figure 143. IRRAS-to-SETS submenu.

This submenu allows us to select from the various types of SETS data to be converted to IRRAS format. We selected the Fault Trees conversion option, and the menu in Figure 144 was displayed on the screen. Had there been no SETS files in the current directory, IRRAS 2.0 would have not put up the next menu but would have given the message

NO SETS files to convert...(extension of '.SET')

Convert From SETS To IRRAS

Option |C| Exit / Convert Fault Tree ← select a Fault Tree below

CCS
ECS
LOSP
FAULT TREES

Note : Use <PgUp> and <PgDn> to display more Fault Trees.

Figure 144. Convert From SETS to IRRAS menu for fault trees.

Figure 144 shows the names of the .SET files in the current directory. Besides the CCS1.SET file, there were also the ECS1.SET and IE-LOSP1.SET files available for conversion to the IRRAS format. To convert a file, the Convert Fault Tree option, C, was selected, the desired file was highlighted, and <Enter> was pressed. IRRAS 2.0 displayed

Converting SETS data file

and began the conversion. When the conversion was completed, IRRAS 2.0 displayed the message

Tree converted - file suffix is .TRE

and gave back the cursor, ready for the next command. This process converted the SETS input file into the IRRAS format. The original SETS input file was kept intact. The output file was called treename.TRE, where treename is the root name of the SETS input file. In order for IRRAS 2.0 to read the converted SETS file, the SETS file needs to be converted into the IRRAS 1.0 file and then converted to 2.0 file. Figure 145 shows the output file, CCS1.TRE.

```

CCS-TOP      OR   DG-B CCS-TRAINS dummyGate1
CCS-TRAINS   AND   CCS-TRAIN-A CCS-TRAIN-B
dummyGate1   OR   CCS-MOV-1 TANK
CCS-TRAIN-A  OR   C-CV-A C-MOV-A DG-A C-PUMP-A
CCS-TRAIN-B  OR   DG-B dummyGate2
dummyGate2   OR   C-MOV-B C-CV-B C-PUMP-B

```

Figure 145. SETS-to-IRRAS conversion output file, CCS1.TRE.

4.4.2 Loading the Converted Files into the Data Base

The file conversion process just described creates IRRAS-compatible logic files in the current directory. To make these files useful, they must be loaded into the IRRAS 2.0 data base through the UTILITY Options module. This module is entered from the IRRAS 2.0 main menu by selecting U - UTILITY Options. On the UTILITY Options menu, select 1 - Version 1.0/2.0 Interface. The Version 1.0/2.0 Interface menu shown in Figure 146 is displayed.

I R R A S - P C
Version 1.0/2.0 Interface

Currently Selected Family —> DEMO FAMILY

E - Exit
S - Select Family

1 - Version 1 to 2

2 - Version 2 to 1

Option... E

Figure 146. IRRAS-PC Version 1.0/2.0 Interface menu.

After checking that the correct family is selected, move the highlight down to the Version 1 to 2 option by using the <Down Arrow> key or press 1 <Enter>. This brings up the Version 1 to 2 options list shown in Figure 147.

Figure 147. Version 1.0 to 2.0 conversion options.

On this menu, select option T - Fault Trees. Selecting this option causes IRRAS 2.0 to display the Convert Fault Tree Version 1.0 to 2.0 menu. This menu is shown in Figure 148.

Figure 148. Convert Fault Tree Version 1.0 --> 2.0 menu (SETS).

All the .TRE files contained in the current directory are listed. Select L - Load Fault Tree, highlight the desired fault tree, and press <Enter> to load the fault tree logic into the IRRAS 2.0 data base. IRRAS 2.0 indicates

Loading tree - please stand by

and indicates completion with

Data successfully loaded...

IRRAS 2.0 is then ready for the next command. The .TRE file remains in the current directory. The loaded fault tree is ready to process (see Section 3.5.1). Reliability data can be added for any basic events not previously in the family, and the new trees can be assigned levels (see Section 3.4 and 3.3.6). All processing, except for graphics, is conducted the same for these trees as for those created in the IRRAS 2.0 system.

More information on the use of the IRRAS Version 1.0/2.0 interface option is provided in Section 4.5.

4.4.3 Converting the Alphanumeric Logic File to Graphics

Converting the SETS input file to the IRRAS format (see Section 4.4.1) allows for analysis of the fault tree but does not provide the graphics for reports and analyst interface. IRRAS 2.0 has the capability to convert the logic file (.TRE) into simple graphics (.DLS file). This is accomplished through the BUILD Fault Trees module.

From the IRRAS 2.0 Main menu, the user selects the BUILD Fault Trees option to get the IRRAS-PC Fault Tree Graphics System menu. On this menu, select the A - Alpha to Graphics option. This brings up the Convert Fault Trees menu as shown in Figure 149.

```

      Convert Fault Trees

Option |C| Exit / Convert Fault Tree  ← select a Fault Tree below

      CCS      CCS      LOSP      FAULT TREES

Note : Use <PgUp> and <PgDn> to display more Fault Trees.
```

Figure 149. Convert Fault Trees menu.

This menu lists all the files with a .TRE suffix in the current directory. To create the graphics for a logic file, the user selects option C - Convert Fault Tree, highlights the desired tree, and presses <Enter>. IRRAS 2.0 begins processing the graphics and indicates

Converting tree - please stand by

When the graphics conversion is completed, IRRAS 2.0 displays

Graphics image created - file suffix is .DLS

and the cursor returns. Thus, if the CCS logic file was highlighted, IRRAS 2.0 created a CCS.DLS file in the current directory. This file contains the graphics drawing representing the logic in the .TRE file. The graphics will have all the gates and basic events, along with their names. Other descriptive text will not be present.

NOTE: A .DLS file in the current directory with the same root name as the .TRE file being converted will be written over. Also, creation of a .DLS file with the same root name as a tree that already exists in the data base will cause the existing tree to be replaced with the newly created graphics. Therefore, it is wise to make sure that the creation of a new .DLS file will not destroy any desired portion of the data base before proceeding with the alpha-to-graphics conversion.

If the new graphics and logic files are for a fault tree that has never been in the data base before, then they must be entered in. This is done by going into the graphics module, calling up the main menu with the mouse or keyboard and exiting (see Sections 3.3.2, 3.3.3, and 3.3.5.3). IRRAS 2.0 will automatically load all fault trees with .TRE files into the data base. These trees can then be assigned levels and processed as any other tree (see Section 3.3.6).

NOTE: While using the BUILD module to load the .TRE files into the data base is fast, it also deletes all the .TRE files from the current directory upon completion. Therefore, be sure to transfer into another directory any .TRE files that you want to keep before going into the graphics.

4.5 IRRAS 1.0/2.0 Interface

One of the major changes from Version 1.0 to Version 2.0 is that the disk files used in Version 1.0 were integrated into a relational data base. To minimize the effects of changing from IRRAS Version 1.0 to Version 2.0, a utility is included in IRRAS 2.0 that automatically converts failure rates, fault tree logic, cut sets, and fault tree graphics from the Version 1.0 format to the Version 2.0 format. In addition, the capability exists to convert Version 2.0 failure rates, fault tree logic, and cut sets back to the Version 1.0 format.

4.5.1 Converting from IRRAS 1.0 to IRRAS 2.0

To convert IRRAS 1.0 data to the IRRAS 2.0 data base, go into the UTILITY Options module and select option V - Version 1.0/2.0 Interface. Figure 150 shows this option on the IRRAS-PC Utilities menu.

Figure 150. IRRAS-PC utilities menu (option V) .

This brings up the IRRAS-PC Version 1.0/2.0 Interface menu shown in Figure 146

After selecting the correct family (see Section 3.3.1), select option 1 - Version 1 to 2. This brings up the Version 1 to 2 conversion options menu shown in Figure 147.

4.5.1.1 Failure Rates. Version 1.0 failure rates are stored in disk files using the format shown in Figure 151. There must be only one rate file for each family, and each file is identifiable by a .RAT extension. In the 1.0 to 2.0 conversion, the file is read into memory and then added to the relational data base for the currently selected family.

Figure 151. Failure rate file format.

To convert the IRRAS 1.0 rate file (FAMILYNAME.RAT) and load it into the IRRAS 2.0 relational data base, select F - Failure

Rates and press <Enter>. The Convert Event Data Version 1.0 ==> 2.0 form, shown in Figure 152, comes on the screen.

Convert Event Data Version 1.0 --> 2.0

Option |L| Exit / Load Event Data ← select a data file below.

Version 1.0 Event Data Files

DEMOFAM

Note : Use <PgUp> and <PgDn> to display more Event Data Files.

Figure 152. Convert Event Data Version 1.0 --> 2.0 form.

This form lists all the .RAT files in the current directory. To load one of these files into the current family, select L - Load Event Data, highlight the desired file name, and press <Enter>. The data will be read into memory and placed in the relational data base under the current family.

NOTE: The .RAT file must be in the exact format of the IRRAS 1.0 output, including the number of significant digits on the failure rates and probabilities. Rate files created by means other than IRRAS 1.0 must be modified to fit this format.

4.5.1.2 Fault Trees. Fault tree logics for Version 1.0 are stored in disk files that have .TRE extensions. Figure 153 shows the Version 1.0 format. When fault tree logic is converted from Version 1.0 to 2.0, a non-zero level tree is created in the data base with the same name as the file that you select. If the converted tree is intended to be a zero level tree, you must go to the MODIFY Data Base module and assign the newly created fault tree a zero level (see Section 3.3.6). Note also that this is a conversion of the alphanumeric representation of the logic only and

no graphics representation exists after completion of this option. If graphics and alphanumeric logic are desired, use the Alpha to Graphics conversion in the BUILD Graphics Trees module instead of this option.

Figure 153. Fault tree logic file format.

To convert a fault tree logic file to the IRRAS 2.0 data base, select T - Fault Trees from the menu shown in Figure 147. This brings up the Convert Fault Tree Version 1.0 $\Rightarrow$ 2.0 form shown in Figure 154.

Convert Fault Tree Version 1.0 → 2.0

Option [L] Exit / Load Fault Tree ← select a fault tree below.

CCS
Version 1.0 Fault Trees

Note : Use <PgUp> and <PgDn> to display more Fault Trees.

Figure 154. Convert Fault Tree Version 1.0 --> 2.0 form.

This form lists all the .TRE files in the current directory. Convert a fault tree logic file by selecting L - Load Fault Tree, highlight the desired logic file, and press <Enter>. The logic file will be read into the data base as a non-zero level fault tree. The fault tree name will be the same as the Version 1.0 logic file name without the extension.

NOTE: Using this option does not remove the .TRE file from the current directory. IRRAS 1.0 logic files may also be entered by entering the graphics module and exiting. This causes all .TRE files to be read into the data base, but it also deletes the .TRE files in the process. Therefore, do not enter the graphics module if there are any .TRE files in the current directory that you do not want destroyed.

4.5.1.3 Cut Sets. Version 1.0 cut sets are stored in disk files that have the same name as the tree logic but with a .CUT extension. Figure 155 shows the format of both quantified and unquantified cut set files. When converting from Version 1.0 to 2.0, the cut set file you select is loaded into memory and then added to the data base under a tree name corresponding to the

selected file name. When events are found in the cut sets that are not already present for the currently selected family, the events are added to the data base.

Figure 155. Cut set file formats.

To convert and load Version 1.0 cut sets into IRRAS 2.0, select C - Cut Sets on the menu shown in Figure 147, and press <Enter>. This brings up the Convert Cut Sets Version 1.0 ==> 2.0 form shown in Figure 156.

Convert Fault Tree Cut Sets							
Option L Exit / Load Cut Sets ← select a cut set file below.							
CCS	ECS	LOSP	Cut Set Files				

Note : Use <PgUp> and <PgDn> to display more Cut Set Files.

Figure 156. Convert Cut Sets Version 1.0 --> 2.0 form.

This form lists all the .CUT files in the current directory. To convert and load one of these files, select L - Load Cut Sets, highlight the desired cut set file, and press <Enter>. The cut sets will be loaded into the data base under a tree with the same name as the Version 1.0 cut set file selected.

4.5.1.4 Graphics. In addition to incorporating a relational data base in Version 2.0, the representation of the graphic images was modified. To process Version 1.0 graphics in Version 2.0, you must use the graphics conversion option. You may convert selected Version 1.0 files or all Version 1.0 files from the current directory. Processing for this option loads a Version 1.0 .DLS file, converts it to the equivalent Version 2.0 representation, deletes the old .DLS file, and replaces it with the new one. An alphanumeric representation of the tree is also generated and stored in a file with the same name and a .TRE extension.

To convert the graphics files, select G - Graphics, highlight the desired .DLS file, and press <Enter>. This brings up the Convert Graphics Version 1.0 ==> 2.0 form shown in Figure 157.

Convert Graphics Version 1.0 → 2.0

Option |C| Exit / Convert graphics ← select a DLS file below.

Graphics Files							
CCS	CCS1	CONN	ECS	ECS1	IE-LOSP	MFRAME	TREE

Note : Use <PgUp> and <PgDn> to display more Fault Trees.

Figure 157. Convert Graphics Version 1.0 --> 2.0 form.

This form lists all the Version 1.0 .DLS files in the current directory. To convert a graphics file to Version 2.0, select C - Convert graphics, highlight the desired fault tree file, and press <Enter>. The .DLS file is read into memory, deleted from the disk, converted to the IRRAS 2.0 format and, written out to the disk as a new .DLS file with the same name as before.

4.5.2 Converting Data from IRRAS 2.0 to IRRAS 1.0

To convert IRRAS 2.0 data to the IRRAS 1.0 format, go into the UTILITY Options module and select option V - Version 1.0/2.0 Interface (Figure 151). This brings up the IRRAS-PC Version 1.0/2.0 Interface menu (Figure 146).

After selecting the correct family (see Section 3.3.1), select option 2 - Version 2 to 1. This brings up the Version 2 to 1 conversion options menu shown in Figure 158.

Figure 158. Version 2.0 to 1.0 conversion options.

4.5.2.1 Failure Rates. During the 2.0 to 1.0 conversion process, failure rate data for the currently selected family are obtained from the data base, formatted as shown in Figure 151, and stored on disk under a name you provide. A .RAT extension is automatically appended.

To convert IRRAS 2.0 failure data to the IRRAS 1.0 format, select F - Failure Rates from the menu in Figure 158, and press <Enter>. This brings up the Convert Event Data Version 2.0 to 1.0 form shown in Figure 159.

Convert Event Data

Option |S| Exit / Store Event Probability Data

Current Family is: DEMO FAMILY

Enter Output File Name: DEMO FAM.RAT

Figure 159. Convert Event Data Version 2.0 --> 1.0 form.

To convert the failure rate data, select S - Store Event Probability Data, and type in the name of the file in which the data are to be saved. The failure rate data for all the basic events in the family will be converted and placed in the file.

4.5.2.2 Fault Trees. Version 2.0 to 1.0 conversion creates a file in the same format as shown in Figure 153. You may choose to store either the logic as represented in the graphics or the modified logic resulting from preprocessing the tree (coalescing of gates and separation of independent subtrees). The name of the newly created file is the first eight characters of the tree name with .TRE appended.

To convert a fault tree logic to the Version 1.0 format, select T - Fault Trees on Figure 158, and press <Enter>. This brings up the Convert Fault Tree Version 2.0 ==> 1.0 form. This form is shown in Figure ?.

Convert Fault Tree Version 2.0 ==> 1.0

Option |E| Exit / Logic / Modified Logic <== select a fault tree below.

Version 2.0 Fault Trees	
CCS	Containment Cooling System fault tree
ECS	Emergency Cooling System fault tree
LOSP	Loss of Offsite Power Initiating Event fault tree

Note : Use <PgUp> and <PgDn> to display more Fault Trees.
 Output file name will be first 8 characters of tree name with ".TRE" extension

Figure 160. Convert Fault Tree Version 2.0 --> 1.0 form.

This form lists all the zero level fault trees in the family. From this menu, the user selects L - Logic or M - Modified logic, highlights the desired fault tree, and presses <Enter>. The logic is converted to the Version 1.0 format and placed in a file in the current directory. The file is given the name of the fault tree selected with the extension .TRE appended.

4.5.2.3 Cut Sets. In the conversion of cut sets from Version 2.0 to 1.0, cut set data for the selected fault tree is extracted from the data base, formatted as shown in Figure 155, and stored on disk in a file given the name of the selected fault tree with the extension .CUT appended.

To convert a cut set list to the Version 1.0 format, select C - Cut Sets on the menu shown in Figure 158 and press <Enter>. This brings up the Convert Cut Sets Version 2.0 ==> 1.0 form shown in Figure 161.

Convert Cut Sets Version 2.0 → 1.0

Option |S| Exit / Store Cut Sets ← select a fault tree below.

Version 2.0 Fault Trees

CCS	Containment Cooling System fault tree
ECS	Emergency Cooling System fault tree
LOSP	Loss of Offsite Power Initiating Event fault tree

Note : Use <PgUp> and <PgDn> to display more Fault Trees.
 Output file name will be the first eight characters
 of the fault tree name with and extension of ".CUT".

Figure 161. Convert Cut Sets Version 2.0 --> 1.0 form.

This form lists all the fault trees that have cut sets in the current family. Select S - Store Cut Sets, highlight the desired cut set list, and press <Enter>. The computer asks

Include Quantification ? (Y/N) |Y|

A Yes response will extract cut sets from the data base, format them as shown in Figure 155, and place them in a file on the disk. The file name will be the fault tree name with the .CUT extension appended.

4.6 Define Constants

Figure 162 shows the IRRAS-PC Utilities menu.

I R R A S - P C
Utilities

E - Exit
 D - Define Constants
 C - Communications
 P - PRA Data Base Interface
 I - IRRAS/SETS Interface
 V - Version 1.0/2.0 Interface
 R - Recover DataBase

Option... D

Figure 162. IRRAS-PC Utilities menu (option D).

The Define Constants option is used to tell the IRRAS 2.0 program what hardware and software configurations it must work with. Figure 163 shows the IRRAS Constants form.

I R R A S C o n s t a n t s

User Information			
User Name	EG&G Idaho, Inc. (INEL)		
Monitor type (0-EGA, 1-CGA)			0
Location type (1-KeyBoard/Tablet/Logitech, 2-Microsoft)			1
Printer has complete IBM character font? (Y/N)			N
Plotter communications port (0=None, 1-Com1, 2-Com2, 3-Com3)			0
Use primary basic event Names ? (Y=primary, N=alternate)			Y
Use ratios in Importance Measures ? (Y=ratios, N=intervals)			Y
Uncertainty			
Random number seed for uncertainty calculations			3571
Sample size			1000
Cut Set Generation			
Cutoff by size? (Y/N)	Y	Size cutoff	6
Cutoff by probability? (Y/N)	Y	Probability cutoff	1.000E-015
Mission time (hours)			2.400E+001
File Locations			
Halo drive and directory path	\HALO\		

Change any of the constant values shown and press the <Enter> key.

Figure 163. IRRAS Constants form (option D).

Each field on the form is accessed by moving the cursor with the arrow keys. The form is divided into four areas:

- o User Information. User identification, hardware configuration information, and basic event name preference.
- o Uncertainty. Defaults for the uncertainty calculations.
- o Cut Set Generation. Defaults for the cut set generation criteria and quantification.
- o File Locations. Paths for the Halo graphics driver and text editor.

Table 2 provides descriptions of the fields and their allowable entries.

FIELD	DESCRIPTION
User Name	36 character user identification (optional) field.
Monitor type	0 - Enhanced graphics monitor and card. 1 - Standard color graphics monitor and board.
Locator type	1 - Keyboard/Tablet/Logitech 2 - Microsoft
Printer has complete IBM character font?	Y - Yes. N - No.
Plotter communications port	0 - No Ploter connected to port. 1 - Plotter connected to Com1 port. 2 - Plotter connected to Com2 port. 3 - Plotter connected to Com3 port.
Use primary basic event names?	Y - Primary name will be used. N - Alternate name will be used.
Use ratio Importance Measures?	Y - Ratios N - Intervals
Random number seed for uncertainty calculations	5 character numeric field indicating the first random number in the seed to be used in the Monte Carlo calculations.
Sample size	6 numeral field indicating the default number of Monte Carlo samples to be run in the uncertainty analyses. Sample size may range from 1 to 999999.
Cutoff by size?	Y - Do not generate fault tree or sequence cut sets containing more basic events than indicated in the Size cutoff field. N - Generate all cut sets for the fault tree or sequence that meet the probability cutoff criteria (if in effect).

Table 2. IRRAS Constants Form Field Descriptions.

FIELD	DESCRIPTION
Size cutoff	The default maximum number of basic events allowed in cut set generation when size cutoff is in effect.
Cutoff by probability?	Y - Do not generate fault tree or sequence cutsets that have a probability less than the cutoff indicated in the probability cutoff field. N - Generate all cut sets that meet the size cutoff criteria (if in effect) regardless of the cut set probability.
Probability cutoff	The default minimum cutset probability allowed in cut set generation when probability cutoff is in effect.
Mission time (hours)	The default mission time to be used in the calculation of basic event probabilities (when appropriate).
Halo drive and directory path	36 character field indicating the drive and path to the Halo graphics that IRRAS 2.0 should use.
Text editor drive and dir path	36 character field indicating the drive and path to the text editor to be used with IRRAS 2.0.

Table 2. (continued).

When the IRRAS Constants form is filled out properly, press <Enter> to send the form to the data base. This will bring up another form, the IRRAS Constants Graphics Information form, show in Figure 164.

I R R A S C o n s t a n t s			
Graphics Information			
Names			
Event Name	EVENT	Name Height	.50
Gate Name	GATE	Line Type	1
Text			
Height	1.00	Justification C	Box Y
N/M Height	1.50	Space Factor	1.40
Toggles		Colors	
Grid	N	Colors ==>	
Blank	Y	Fill Color	X
Show Name	Y	Name Color	X
Show Text	Y	Line Color	X
MultiPick	Y	Text Color	X
Change any of the constant values shown and press the <Enter> key.			

Figure 164. IRRAS Constants Graphics Information form (option D).

This form is divided into four areas:

- o Names. Event name default, gate name default, name height, and line type.
- o Text. Text height, N/M OR gate designation height, text alignment (left, center, right justification), space factor (spacing between lines of text), and box (very small names to appear as a box versus words).
- o Toggles. Turn on or off the following parameters: grid, blank, show name, show text, and multi-pick.
- o Colors. Sets the default colors for fill, name, line, and text.

Table 3 provides descriptions of the fields and their allowable entries.

FIELD	DESCRIPTION
Event Name	10 character field for the fault tree basic event default name. The default name is followed by a sequential number for each separate event, e.g., EVENT1, EVENT2, etc.
Gate Name	10 character field for the fault tree gate default name. The default name is followed by a sequential number for each separate event, e.g., GATE1, GATE2, etc.
Name Height	The height of the event and gate names. This is a number between 0.01 and 66.00, where 66.00 represents the full 66 lines from the top of the screen to the bottom.
Line Type	1 - Solid line. 2 - Dashed line. 3 - Dotted line.
Text Height	The height of the text to be written. This is a number between 0.01 and 66.00 (see Name Height).
N/M Height	The height of the numbers on an N/M OR gate showing the N and M values. This is a number between 0.01 and 66.00.
Justification	Text justification. L - Left justified. C - Centered. R - Right justified.
Space Factor	The spacing between lines of text written consecutively. This is a number between 1.00 and 10.00. A value of 1.00 will cause the top of the following line of text to touch the bottom of the preceeding line.

Table 3. IRRAS Constants Graphics Information Form Field Descriptions.

FIELD	DESCRIPTION
Box	Y - A box will be drawn in place of a gate or event name should the scale make name text unreadable. N - The name text will always appear regardless of the scale used.
Grid	Yes/No toggle turning a reference grid on/off.
Blank	Y - The immediate area surrounding a gate or event name will be blanked out. N - The gate and event names will be written over any lines drawn in the areas for the names.
Show Name	Yes/No toggle turning on/off the display of event and gate names when the SHOW command is used.
Show Text	Yes/No toggle turning on/off the display of descriptive text when the SHOW command is used.
MultiPick	Yes/No toggle turning on/off MultiPick.
Fill Color	An X under the desired color selects the default color for filling in shapes.
Name Color	An X under the desired color selects the default color for displaying names.
Line Color	An X under the desired color selects the default color for drawing lines.
Text Color	An X under the desired color selects the default color for writing text.

Table 3. (continued).

4.7 Communicating with a Mainframe Computer

IRRAS 2.0 is capable of communicating with a mainframe computer through the use of a terminal emulation communications processor called CONVERSE. CONVERSE is designed to allow convenient access to another computer system. It emulates the VT100 terminal. Another library named COMMA60 provides ADDS Viewpoint 60 emulation.

CONVERSE provides the capability of defining the various communication parameters, such as baud rate (speed of transmission), parity, and others. It also provides an offline editing capability along with up to 375 lines (15 pages) of scrollable memory. The additional capability to give commands directly or from a command file allows a user to direct CONVERSE to perform required functions.

In order to communicate with the mainframe, select the UTILITIES option under IRRAS 2.0 main menu. This brings up the UTILITIES menu, as shown in Figure 162. Select C - Communications.

WARNING: Do not select C - Communications unless you are connected up with the mainframe computer. Selecting C - Communications without being connected to the mainframe will cause the IRRAS 2.0 program to lock-up. If this should occur, do a soft boot (Cntrl, Alt, Del pressed simultaneous).

For a detailed description of the communications options, please refer to Appendix E.

4.8 More on the IRRAS/SETS Interface

Section 4.4 described how to convert SETS input logic files to the IRRAS format. This is only half of the IRRAS/SETS interface capabilities. This section discusses the other half, converting IRRAS logic files to the SETS format.

Although IRRAS-PC is a powerful tool, it is realized that there will be times when the models and analysis needs will exceed the limits of a personal computer. For these situations, IRRAS 2.0 provides a means of creating logic files compatible with the SETS mainframe computer code.

The following example problem illustrates the use of this feature. A fault tree logic model was created in IRRAS 2.0 using the graphics capabilities and is to be processed on the mainframe computer using SETS.

The steps required to do this are:

1. Create an IRRAS format logic file in the current directory.
2. Convert the IRRAS logic file to the SETS format.
3. Send the SETS file to the mainframe and process.

Step 1 is described in Section 4.5 and Step 3 is detailed in Section 4.7. Step 2 is described below.

After creating the IRRAS logic file using the IRRAS 1.0/2.0 Interface option, select the IRRAS/SETS Interface option on the

Utilities menu by pressing I and <Enter>. This brings up the IRRAS-PC IRRAS to SETS Interface menu as shown in Figure 165.

I R R A S - P C Irras/Sets Interface	
Currently Selected Family —> DEMO FAMILY	
E - Exit S - Select Family	
I - Irras to Sets	X - Sets to Irras
Option... I	

Figure 165. IRRAS-PC IRRAS-to-SETS Interface menu (option I).

On this menu, select the IRRAS-to-SETS conversion option to get the IRRAS to SETS submenu shown in Figure 166. This submenu allows the user to select either failure rates or fault trees to be converted to SETS format.

I R R A S - P C
 Irras/Sets Interface

Currently Selected Family —> DEMO FAMILY

E - Exit
 S - Select Family

I - Irras to Sets

X - Sets to Irras

E - Exit
 F - Failure Rates
 T - Fault Trees

Option ... E

Figure 166. IRRAS-to-SETS submenu.

If we select the Failure Rates option, the menu shown in Figure 167 will be displayed. This menu allows us to write the IRRAS basic event failure data to a Sets value block formatted file.

Convert Event Data

Option [S] Exit / Store Event Probability Data

Current Family is :DEMO FAMILY

Enter Output File Name : DEMOFAM .RAT

Figure 167. IRRAS-to-SETS Convert Event Data submenu.

If we select Fault Trees, the menu shown in Figure 168 will be displayed.

I R R A S t o S E T S C o n v e r s i o n

Currently Selected Family —> DEMO FAMILY

Option |C| Exit / Convert Fault Tree <— Select Fault Tree Below

FAULT TREES

CCS	Containment Cooling system fault tree
ECS	Emergency Cooling system fault tree
LOSP	Loss of Offsite Power initiating event fault tree

Note : Use <PgUp> and <PgDn> to display more fault trees.

Figure 168. IRRAS-to-SETS Fault Tree menu.

This menu lists all the fault trees in the family regardless of their level assignments. The fault tree for this example is MFRAME. Figure 169 shows a listing of the IRRAS logic for MFRAME.

CCS-TOP	OR	DG-B, ECS-TRAINS, DG-A, CCS-TRAINS, dummyGate1
CCS-TRAINS	AND	CCS-TRAIN-A, CCS-TRAIN-B
ECS-TRAINS	AND	ECS-TRAIN-B, ECS-TRAIN-A
dummyGate1	OR	TANK, CCS-MOV-1, ECS-MOV-1
CCS-TRAIN-A	OR	DG-A, dummyGate2
CCS-TRAIN-B	OR	DG-B, dummyGate3
ECS-TRAIN-A	OR	DG-A, dummyGate4
ECS-TRAIN-B	OR	DG-B, dummyGate5
dummyGate2	OR	C-MOV-A, C-CV-A, C-PUMP-A
dummyGate3	OR	C-CV-B, C-PUMP-B, C-MOV-B
dummyGate4	OR	E-CV-A, E-PUMP-A, E-MOV-A
dummyGate5	OR	E-MOV-B, E-CV-B, E-PUMP-B

Figure 169. MFRAME.TRE logic file.

To prepare MFRAME for analysis on the mainframe computer using SETS, select option C for Convert Fault Tree, highlight MFRAME, and press <Enter>. IRRAS 2.0 rapidly displays several status messages while processing and then displays

Tree converted - file suffix is .SET

A SETS input file called MFRAME.SET now exists in the current directory that can be sent to the mainframe computer for analysis. Figure 170 is a listing of this file.

```

OG$ CCS-TOP.          IN$ DG-B, ECS-TRAINS, DG-A, CCS-TRAINS, dummyGate1.
AG$ CCS-TRAINS.       IN$ CCS-TRAIN-A, CCS-TRAIN-B.
                        OUT$ CCS-TOP.
AG$ ECS-TRAINS.       IN$ ECS-TRAIN-B, ECS-TRAIN-A.
                        OUT$ CCS-TOP.
OG$ dummyGate1.       IN$ TANK, CCS-MOV-1, ECS-MOV-1.
                        OUT$ CCS-TOP.
OG$ CCS-TRAIN-A.      IN$ DG-A, dummyGate2.
                        OUT$ CCS-TRAINS.
OG$ CCS-TRAIN-B.      IN$ DG-B, dummyGate3.
                        OUT$ CCS-TRAINS.
OG$ ECS-TRAIN-A.      IN$ DG-A, dummyGate4.
                        OUT$ ECS-TRAINS.
OG$ ECS-TRAIN-B.      IN$ DG-B, dummyGate5.
                        OUT$ ECS-TRAINS.
OG$ dummyGate2.       IN$ C-MOV-A, C-CV-A, C-PUMP-A.
                        OUT$ CCS-TRAIN-A.
OG$ dummyGate3.       IN$ C-CV-B, C-PUMP-B, C-MOV-B.
                        OUT$ CCS-TRAIN-B.
OG$ dummyGate4.       IN$ E-CV-A, E-PUMP-A, E-MOV-A.
                        OUT$ ECS-TRAIN-A.
OG$ dummyGate5.       IN$ E-MOV-B, E-CV-B, E-PUMP-B.
                        OUT$ ECS-TRAIN-B.
BE$ C-CV-A. OUT$ dummyGate2.
BE$ C-CV-B. OUT$ dummyGate3.
BE$ C-MOV-A. OUT$ dummyGate2.
BE$ C-MOV-B. OUT$ dummyGate3.
BE$ C-PUMP-A. OUT$ dummyGate2.
BE$ C-PUMP-B. OUT$ dummyGate3.
BE$ CCS-MOV-1. OUT$ dummyGate1.
BE$ DG-A. OUT$ CCS-TOP, CCS-TRAIN-A, ECS-TRAIN-A.
BE$ DG-B. OUT$ CCS-TOP, CCS-TRAIN-B, ECS-TRAIN-B.
BE$ TANK. OUT$ dummyGate1.
BE$ E-CV-A. OUT$ dummyGate4.
BE$ E-CV-B. OUT$ dummyGate5.
BE$ E-MOV-A. OUT$ dummyGate4.
BE$ E-MOV-B. OUT$ dummyGate5.
BE$ E-PUMP-A. OUT$ dummyGate4.
BE$ E-PUMP-B. OUT$ dummyGate5.
BE$ ECS-MOV-1. OUT$ dummyGate1.

```

Figure 170. SETS input file MFRAME.SET.

4.9 Recovering the Data Base

(Guidance will be provided on when to recover the data base and when not to recover it.)

To recover the data base, select R - Recover Data Base on the IRRAS-PC Utilities menu (Figure 162). Since the data base can be rather extensive, IRRAS 2.0 provides the warning shown in Figure 171.

Figure 171. Recover Data Base warning.

If you do not want to continue, just press <Enter>. If you want to go ahead and recover the data base, press Y and <Enter>. IRRAS 2.0 will go through the entire data base and restructure the data base and re-index the data.

NOTE: Before recovering the data base, it is suggested that the user backup the IRRAS.DFL file and the data files (*.DAT, *.IDX, and *.BLK) as a precaution against loss.

4.10 Backing Up/Transferring the Data Base

The entire IRRAS 2.0 data base is contained in three types of files. These are data files (*.DAT), index files (*.IDX), and block files (*.BLK). The graphics display list files (*.DLS) are not considered part of the data base because they can be extracted from the other types of data files. Therefore, to back up your data base or transfer it to another location, copy the *.DAT, *.IDX, and *.BLK files to your storage or transfer media. The number of individual files involved depends upon what operations have been performed with the data base.

Other files that you may consider backing up are the raster image files (*.RAS), because they take time to reproduce, and any report files that may have been created in the course of your work.

4.11 Editing the Fault Trees

Once we have generated the fault trees, we may find it necessary to make some minor or even major changes. To make these changes we could go back to the Build menu, make the changes to the tree or sequence, and regenerate the fault trees. However, the Fault Tree Editor will allow us to make essential changes directly to the fault trees using gate and event names. [Note : these changes will not be reflected in the graphics.]

From the Fault Tree Analysis System menu (see Figure 30), we selected the Fault Tree Editor option, which brought up the Select a Fault Tree menu, shown in Figure 172. We positioned the cursor over the CCS fault tree and pressed the enter key. (Since F is the default menu option, we directly selected our fault tree.) This brought up the menu for the Fault Tree Editor and displayed our fault trees, shown in Figure 173. We noted that gate inputs were colored with cyan letters and event inputs were colored with white letters.

FAULT TREE
EDITOR

Currently Selected Family => DEMO FAMILY

Option |F| Exit / Fault Tree Editor <=== Select a Fault Tree Below

FAULT	TREES
CCS	Containment Cooling System fault tree
ECS	Emergency Cooling System fault tree
LOSP	Loss of Offsite Power Initiating Event fault tree

Note : Use <PgUp> and <PgDn> to display more fault trees.

Figure 172. Select a Fault Tree menu.

FAULT TREE EDITOR				
Exit / Add / Modify / Delete / Locate / Next / Previous Insert / Replace / Search / Options / Undo Delete / Esc for more help				
Gate Name	Type	Inputs		
C-MOV-1-FAILS	OR	C-MOV-1	DG-B	
CCS-SUPPLY	OR	C-MOV-1-FAILS	TANK	
CCS-TOP	OR	CCS-SUPPLY	CCS-TRAINS	
CCS-TRAIN-A	OR	C-CV-A	C-MOV-A	DG-A
		C-PUMP-A		
CCS-TRAIN-B	OR	C-CV-B	C-MOV-B	C-PUMP-B
		DG-B		
CCS-TRAINS	AND	CCS-TRAIN-A	CCS-TRAIN-B	

Figure 173. Fault Tree Editor menu and window.

The Fault Tree Editor has 13 menu options, each of which require only a single key stroke to initiate the command:

- o Exit. End your editing session.
- o Add. Allow addition of an input in an existing gate or addition of a gate, the gate type, and up to 45 inputs.
- o Modify. Permit the user to make a change to a gate name, gate type, or input name.
- o Delete. Remove an input or an entire gate, including its inputs and any reference to the deleted gate in other gates.
- o Locate. Find the first occurrence of a given string.
- o Next. Find the next occurrence of a located string.
- o Previous. Find the previous occurrence of a located string.
- o Insert. Add an input to each gate containing an input matching a search string.

- o Replace. Replace all inputs matching a search string with a given input name.
- o Search. Look for the first occurrence of each letter as it is typed.
- o Options. Set the veto and help defaults for the Fault Tree Editor.
- o Undo Delete. Reverse the results of a deleted input. Note that Undo does not reverse deleted gates.
- o Esc. Toggle between the command menu and the cursor help menu.

We pressed the Esc key and noticed that the command menu was replaced by the cursor help menu, as shown in Figure 174. Here we saw that to move the cursor about the editing window we used the keyboard cursor pad. The arrow keys moved us one field in each respective direction. The Home key places the cursor in the upper right corner of the window, and the End key moves the cursor to the bottom right. The PgUp key moved the window up one page (15 lines). "PgDn" moved the window down one page. Ctrl-Home and Ctrl-End moved us to the top and bottom of the file.

FAULT TREE EDITOR				
< >Cursor Up < >Cursor Down < >Cursor Left < >Cursor Right <End>End of Page <Home>Top of Page <PgUp>Page Up <PgDn>Page Down <Ctrl-PgUp>Top of File <Ctrl-PgDn>End of File <Esc> for more				
Gate Name	Type	Inputs		
C-MOV-1-FAILS	OR	C-MOV-1	DG-B	
CCS-SUPPLY	OR	C-MOV-1-FAILS	TANK	
CCS-TOP	OR	CCS-SUPPLY	CCS-TRAINS	
CCS-TRAIN-A	OR	C-CV-A	C-MOV-A	DG-A
		C-PUMP-A		
CCS-TRAIN-B	OR	C-CV-B	C-MOV-B	C-PUMP-B
		DG-B		
CCS-TRAINS	AND	CCS-TRAIN-A	CCS-TRAIN-B	

Figure 174. Fault Tree Editor cursor help menu.

FAULT TREE EDITOR

New Gate Addition

Note : for more than 45 inputs, enter with "Add" in input field.
 Use <Tab> to move between fields, <Enter> when done, <Esc> to abort

Gate Name	Type	Inputs		
CCS-NEW	AND	DG-A	DG-B	
C-MOV-1-FAILS	OR	C-MOV-1	DG-B	
CCS-SUPPLY	OR	C-MOV-1-FAILS	TANK	
CCS-TOP	OR	CCS-SUPPLY	CCS-TRAINS	
CCS-TRAIN-A	OR	C-CV-A	C-MOV-A	DG-A
		C-PUMP-A		
CCS-TRAIN-B	OR	C-CV-B	C-MOV-B	C-PUMP-B
		DG-B		
CCS-TRAINS	AND	CCS-TRAIN-A	CCS-TRAIN-B	

Figure 176. Completion of a Gate Addition.

Next we demonstrated an addition of an input into a gate. We moved the cursor into the first input column and row so that the cursor was over the input name DG-A and pressed the A key, as before. The field location cleared, and we were left with a clean area to make the entry TANK. Then we used the <Enter> key to make the addition complete, and the window was updated as shown in Figure 177.

FAULT TREE EDITOR

Add Input Name
New names will be automatically added to the gate or event list.
Use <Enter> when done, and <Esc> to abort

Use <Enter> when done, and <Esc> to abort

Gate Name	Type	Inputs		
CCS-NEW	AND	TANK	DG-A	DG-B
C-MOV-1-FAILS	OR	C-MOV-1	DG-B	
CCS-SUPPLY	OR	C-MOV-1-FAILS	TANK	
CCS-TOP	OR	CCS-SUPPLY	CCS-TRAINS	
CCS-TRAIN-A	OR	C-CV-A	C-MOV-A	DG-A
		C-PUMP-A		
CCS-TRAIN-B	OR	C-CV-B	C-MOV-B	C-PUMP-B
		DG-B		
CCS-TRAINS	AND	CCS-TRAIN-A	CCS-TRAIN-B	

Figure 177. Addition of input to a gate.

To demonstrate what happens if an input being entered is unknown, we positioned the cursor to the last gate CCS-TRAINS in the third input column and pressed <A>, followed by CCS-NEW-G2. We were asked if CCS-NEW-G2 is a gate or an event, as shown in Figure 178. We entered <G> for gate. Figure 179 shows the result of the new input as well as the automatic entry of the gate.

FAULT TREE EDITOR				
Gate or Event?				
Enter G, E, or A =>				
Use <G> for Gate, <E> for Event, or <A> for abort then <enter>.				
Gate Name	Type	Inputs		
CCS-NEW	AND	TANK	DG-A	DG-B
C-MOV-1-FAILS	OR	C-MOV-1	DG-B	
CCS-SUPPLY	OR	C-MOV-1-FAILS	TANK	
CCS-TOP	OR	CCS-SUPPLY	CCS-TRAINS	
CCS-TRAIN-A	OR	C-CV-A	C-MOV-A	DG-A
		C-PUMP-A		
CCS-TRAIN-B	OR	C-CV-B	C-MOV-B	C-PUMP-B
		DG-B		
CCS-TRAINS	AND	CCS-TRAIN-A	CCS-TRAIN-B	CCS-NEW-G2

Figure 178. Gate or Event inquiry.

FAULT TREE EDITOR				
Exit / Add / Modify / Delete / Locate / Next / Previous Insert / Replace / Search / Options / Undo Delete / Esc for more help				
Gate Name	Type	Inputs		
CCS-NEW	AND	TANK	DG-A	DG-B
C-MOV-1-FAILS	OR	C-MOV-1	DG-B	
CCS-SUPPLY	OR	C-MOV-1-FAILS	TANK	
CCS-TOP	OR	CCS-SUPPLY	CCS-TRAINS	
CCS-TRAIN-A	OR	C-CV-A	C-MOV-A	DG-A
		C-PUMP-A		
CCS-TRAIN-B	OR	C-CV-B	C-MOV-B	C-PUMP-B
		DG-B		
CCS-TRAINS	AND	CCS-TRAIN-A	CCS-TRAIN-B	CCS-NEW-G2
CCS-NEW-G2	????			

Figure 179. Entering an unknown Gate.

We positioned to the new gate CCS-NEW-G2 in the first input column to demonstrate entering an unknown event. We typed the event name DG-C and received the gate event question. We entered <E>, and the event was automatically added to a temporary event list and inserted into the gate. We repeated the process to enter DG-D to complete the unknown gate, as shown in Figure 180. [Note: the temporary event list is added to the event file if we choose to keep our changes at the end of the editing session.]

FAULT TREE EDITOR				
Exit / Add / Modify / Delete / Locate / Next / Previous Insert / Replace / Search / Options / Undo Delete / Esc for more help				
Gate Name	Type	Inputs		
CCS-NEW	AND	TANK	DG-A	DG-B
C-MOV-1-FAILS	OR	C-MOV-1	DG-B	
CCS-SUPPLY	OR	C-MOV-1-FAILS	TANK	
CCS-TOP	OR	CCS-SUPPLY	CCS-TRAINS	
CCS-TRAIN-A	OR	C-CV-A	C-MOV-A	DG-A
		C-PUMP-A		
CCS-TRAIN-B	OR	C-CV-B	C-MOV-B	C-PUMP-B
		DG-B		
CCS-TRAINS	AND	CCS-TRAIN-A	CCS-TRAIN-B	CCS-NEW-G2
CCS-NEW-G2	????	DG-C	DG-D	

Figure 180. Entering unknown events.

To show how to change the name of an input, we positioned the cursor over the input DG-A in the second input column and pressed the <M> key. This brought up the Modify Name message, and the cursor was placed at the beginning of the field. We entered CCS-SUPPLY and the <Enter> key to terminate. Also we changed the ???? in the type column by positioning the cursor over the ????, pressing the M key, and typing AND. Figure 181 shows the result of the changes.

FAULT TREE EDITOR				
Modify				
Use Insert, Delete, and arrow keys to modify high-lighted text.				
Use <Enter> when done, and <Esc> to abort				
Gate Name	Type	Inputs		
CCS-NEW	AND	TANK	CCS-SUPPLY	DG-B
C-MOV-1-FAILS	OR	C-MOV-1	DG-B	
CCS-SUPPLY	OR	C-MOV-1-FAILS	TANK	
CCS-TOP	OR	CCS-SUPPLY	CCS-TRAINS	
CCS-TRAIN-A	OR	C-CV-A	C-MOV-A	DG-A
		C-PUMP-A		
CCS-TRAIN-B	OR	C-CV-B	C-MOV-B	C-PUMP-B
		DG-B		
CCS-TRAINS	AND	CCS-TRAIN-A	CCS-TRAIN-B	CCS-NEW-G2
CCS-NEW-G2	AND	DG-C	DG-D	

Figure 181. Modification of a gate, type, or input.

Next we observed how the three commands Locate, Next, and Previous functioned. By typing the <L> key, the prompt for the Locate command appeared, and we then entered the string ??-* as shown in Figure 182. The cursor then moved to the first input name whose third character was the "-", highlighting the name DG-B in the first gate. (Actually we could have entered in the name of the input which we were looking for; however, the Fault Tree Editor will allow for wild card search strings. The "*" matches any characters for any length, and the "?" matches any single character in the position where it is found.) Note that the locate command will begin its search in either the gate names or input names depending on the position of the cursor when the command is given. In this case, none of the gate names contained a "-" as the second character, so the input DG-B would have been found regardless of the initial cursor position. Had the locate string been "CCS-*,", then if the cursor was positioned in the gate column, CCS-NEW would have been found; or if the cursor was positioned in the input columns, CCS-SUPPLY would have been found.

FAULT TREE EDITOR				
Locate				
Enter the Gate or Input name to locate => ??-*				
Use <Enter> when done				
Gate Name	Type	Inputs		
CCS-NEW	AND	TANK	CCS-SUPPLY	DG-B
C-MOV-1-FAILS	OR	C-MOV-1	DG-B	
CCS-SUPPLY	OR	C-MOV-1-FAILS	TANK	
CCS-TOP	OR	CCS-SUPPLY	CCS-TRAINS	
CCS-TRAIN-A	OR	C-CV-A	C-MOV-A	DG-A
		C-PUMP-A		
CCS-TRAIN-B	OR	C-CV-B	C-MOV-B	C-PUMP-B
		DG-B		
CCS-TRAINS	AND	CCS-TRAIN-A	CCS-TRAIN-B	CCS-NEW-G2
CCS-NEW-G2	AND	DG-C	DG-D	

Figure 182. Locate a gate or input

The Next command was demonstrated when we pressed the key <N> and noticed that the cursor moved to the next occurrence of our search string. Finally, we typed <P>, and the cursor moved back to the previous input name. This indicated that Locate searches for the first occurrence of a given search string, beginning from the top of the file. Also, Next searches for the next occurrence of the search string, and Previous searches for the previous occurrence of the search string.

Next we demonstrated the Search command to see how it differs from Locate. By entering the <S> key, we were asked to enter in a search string, as shown in Figure 183.

FAULT TREE EDITOR				
Speed Search				
Enter the input name to be search for =>				
Use <Enter> when done				
Gate Name	Type	Inputs		
CCS-NEW	AND	TANK	CCS-SUPPLY	DG-B
C-MOV-1-FAILS	OR	C-MOV-1	DG-B	
CCS-SUPPLY	OR	C-MOV-1-FAILS	TANK	
CCS-TOP	OR	CCS-SUPPLY	CCS-TRAINS	
CCS-TRAIN-A	OR	C-CV-A	C-MOV-A	DG-A
		C-PUMP-A		
CCS-TRAIN-B	OR	C-CV-B	C-MOV-B	C-PUMP-B
		DG-B		
CCS-TRAINS	AND	CCS-TRAIN-A	CCS-TRAIN-B	CCS-NEW-G2
CCS-NEW-G2	AND	DG-C	DG-D	

Figure 183. Search inputs for a string.

Upon entering the string DG-A, we noticed that the cursor moved as soon as we had typed the letter D. This command is called Speed Search, and it attempts to find a match for the search string as each character is entered. After the letter A was entered, we then noticed that the search for this input was found. Unlike Locate, which begins the search at the beginning of the data, Search begins at the top of the current window. Note that the Next and Previous commands can also be used after a Search command completes. For example, assume we were in the middle of fault tree which contained about 400 gates using Next to search for occurrences of a certain input name when we decided to change the search string. Rather than using Locate to change the string and search back down to where we are positioned (remember that Locate begins at the beginning of the data), we can use Search to look for the occurrence of the new search string. The command Next can then be used to look for this new search string beginning at the top of the window.

To demonstrate the Insert and the Replace commands, we entered the key I, and the insert prompt appeared just below the title bar. We entered in the search string (TANK) and the new input name (C-PUMP-A) which was to be added, as shown in Figure 184.

FAULT TREE EDITOR				
Find and Insert				
Find TANK and Insert C-PUMP-A				
Use <Tab> to move between fields, <Enter> when done and, <Esc> for help				
Gate Name	Type	Inputs		
CCS-NEW	AND	TANK	CCS-SUPPLY	DG-B
C-MOV-1-FAILS	OR	C-MOV-1	DG-B	
CCS-SUPPLY	OR	C-MOV-1-FAILS	TANK	
CCS-TOP	OR	CCS-SUPPLY	CCS-TRAINS	
CCS-TRAIN-A	OR	C-CV-A	C-MOV-A	DG-A
		C-PUMP-A		
CCS-TRAIN-B	OR	C-CV-B	C-MOV-B	C-PUMP-B
		DG-B		
CCS-TRAINS	AND	CCS-TRAIN-A	CCS-TRAIN-B	CCS-NEW-G2
CCS-NEW-G2	AND	DG-C	DG-D	

Figure 184. Beginning a Find and Insert.

At the completion of the command, we saw that at each occurrence of the search string the new input was added to that gate, as shown in Figure 185. The Insert command allowed us to search globally for an occurrence of a search string and insert an input.

FAULT TREE EDITOR

Find and Insert
Find TANK and Insert C-PUMP-A
Use <Tab> to move between fields, <Enter> when done and, <Esc> for help

Gate Name	Type	Inputs		
CCS-NEW	AND	TANK	C-PUMP-A	CCS-SUPPLY
		DG-B		
C-MOV-1-FAILS	OR	C-MOV-1	DG-B	
CCS-SUPPLY	OR	C-MOV-1-FAILS	TANK	C-PUMP-A
CCS-TOP	OR	CCS-SUPPLY	CCS-TRAINS	
CCS-TRAIN-A	OR	C-CV-A	C-MOV-A	DG-A
		C-PUMP-A		
CCS-TRAIN-B	OR	C-CV-B	C-MOV-B	C-PUMP-B
		DG-B		
CCS-TRAINS	AND	CCS-TRAIN-A	CCS-TRAIN-B	CCS-NEW-G2
CCS-NEW-G2	AND	DG-C	DG-D	

Figure 185. Completion of a Find and Insert.

Next we selected the R and the Replace prompt was shown, as in Figure 186.

FAULT TREE EDITOR

Find
Find and Replace

Replace with

Use <Tab> to move between fields, <Enter> when done and, <Esc> for help

Gate Name	Type	Inputs		
CCS-NEW	AND	TANK DG-B	C-PUMP-A	CCS-SUPPLY
C-MOV-1-FAILS	OR	C-MOV-1	DG-B	
CCS-SUPPLY	OR	C-MOV-1-FAILS	TANK	C-PUMP-A
CCS-TOP	OR	CCS-SUPPLY	CCS-TRAINS	
CCS-TRAIN-A	OR	C-CV-A C-PUMP-A	C-MOV-A	DG-A
CCS-TRAIN-B	OR	C-CV-B DG-B	C-MOV-B	C-PUMP-B
CCS-TRAINS	AND	CCS-TRAIN-A	CCS-TRAIN-B	CCS-NEW-G2
CCS-NEW-G2	AND	DG-C	DG-D	

Figure 186. Beginning of a Find and Replace.

We entered DG-? for the search string and C-MOV-A for the replacement string. Note that gate CCS-TRAIN-A contained both DG-A and C-MOV-A. Since the replacement of DG-A would cause a duplicate input, the operation was not done in that gate. The operation was completed as shown in Figure 187.

FAULT TREE EDITOR

Find and Replace

Find DG-?
Replace with C-MOV-A

Use <Tab> to move between fields, <Enter> when done and, <Esc> for help

Gate Name	Type	Inputs		
CCS-NEW	AND	TANK	C-PUMP-A	CCS-SUPPLY
C-MOV-1-FAILS	OR	C-MOV-A	C-MOV-A	
CCS-SUPPLY	OR	C-MOV-1	TANK	C-PUMP-A
CCS-TOP	OR	C-MOV-1-FAILS	CCS-TRAINS	
CCS-TRAIN-A	OR	CCS-SUPPLY	C-MOV-A	DG-A
		C-CV-A		
CCS-TRAIN-B	OR	C-PUMP-A	C-MOV-B	C-PUMP-B
		C-CV-B		
CCS-TRAINS	AND	C-MOV-A	CCS-TRAIN-B	CCS-NEW-G2
CCS-NEW-G2	AND	CCS-TRAIN-A	DG-D	
		DG-C		

Figure 187. Completion of a Find and Replace.

When either the Insert or Replace prompts are on the screen the <Esc> key causes a help message to appear, as shown in Figure 188.

This operation uses two key names, the first name may be a wild card string which may include any number of "*" or "?". The "*" is non-positional and length "don't care." The "?" is a non-positional single length character "don't care." The second name can have no wild card characters, and will be used exactly as typed. For example the name string "A*B*" would match any name beginning with an "A:" and containing a "B: anywhere within that name. And "A?B*" would match any name beginning with "A" and having "B" as the third character in the name. Many other combinations are possible, however the second name will overwrite or add the the name matched within the inputs.

Use Esc to end help

Figure 188. Help Message for Find and Insert/Replace.

Next we wished to show how the Delete function operates. The Delete command is like the Add and Modify commands in that it is column dependent. If the cursor is in the gate column, then the entire gate under the cursor is removed from the data. Diversely, if the cursor is in any of the input columns, the input under the cursor is deleted. Note that the delete veto discussed below applies only to the case where a gate is being deleted and that Undo Delete works only with inputs and not gates. We demonstrated the delete by moving the cursor over the gate name CCS-TRAIN-A and pressing the D key. The gate CCS-TRAIN-A was removed from the data including the reference to CCS-TRAIN-A in gate CCS-TRAINS, as shown in Figure 189.

FAULT TREE EDITOR				
Exit / Add / Modify / Delete / Locate / Next / Previous				
Insert / Replace / Search / Options / Undo Delete / Esc for more help				
Gate Name	Type	Inputs		
CCS-NEW	AND	TANK	C-PUMP-A	CCS-SUPPLY
C-MOV-1-FAILS	OR	C-MOV-A	C-MOV-A	
CCS-SUPPLY	OR	C-MOV-1	TANK	C-PUMP-A
CCS-TOP	OR	C-MOV-1-FAILS	CCS-TRAINS	
CCS-TRAIN-B	OR	CCS-SUPPLY	C-MOV-B	C-PUMP-B
		C-CV-B		
CCS-TRAINS	AND	C-MOV-A	CCS-NEW-G2	
CCS-NEW-G2	AND	CCS-TRAIN-B	DG-D	
		DG-C		

Figure 189. Completion of a Delete Gate.

To demonstrate the deletion of an input, we positoned the cursor over TANK in the top of the first input column and pressed the <D> key. The input TANK was deleted, as shown in Figure 190.

FAULT TREE EDITOR				
Exit / Add / Modify / Delete / Locate / Next / Previous Insert / Replace / Search / Options / Undo Delete / Esc for more help				
Gate Name	Type	Inputs		
CCS-NEW	AND	C-PUMP-A	CCS-SUPPLY	
		C-MOV-A		
C-MOV-1-FAILS	OR	C-MOV-1	C-MOV-A	C-PUMP-A
CCS-SUPPLY	OR	C-MOV-1-FAILS	TANK	
CCS-TOP	OR	CCS-SUPPLY	CCS-TRAINS	
CCS-TRAIN-B	OR	C-CV-B	C-MOV-B	C-PUMP-B
		C-MOV-A		
CCS-TRAINS	AND	CCS-TRAIN-B	CCS-NEW-G2	
CCS-NEW-G2	AND	DG-C	DG-D	

Figure 190. Completion of a Delete Input.

Next we demonstrated the Undo Delete by pressing the <U> key, which restored the screen and data as shown in Figure 189. This is the functionality of the Undo Delete. However, it will only work precisely after a deletion has occurred.

The Options command gives us the ability to set a few important defaults associated with the Fault Tree Editor. We illustrated this by typing the O key on the keyboard. The Options menu appeared underneath the title bar, as shown in Figure 191. The settings noted are the defaults which are preset when entering the editor.

FAULT TREE EDITOR				
Options				
Veto <ON> =>N Delete Veto <ON> =>Y Help <ON> =>Y				
Use <Tab> to move between fields, <Enter> when done, and <Esc> for help				
Gate Name	Type	Inputs		
CCS-NEW	AND	TANK	C-PUMP-A	CCS-SUPPLY
C-MOV-1-FAILS	OR	C-MOV-A	C-MOV-A	
CCS-SUPPLY	OR	C-MOV-1	TANK	C-PUMP-A
CCS-TOP	OR	C-MOV-1-FAILS	CCS-TRAINS	
CCS-TRAIN-B	OR	CCS-SUPPLY	C-MOV-B	C-PUMP-B
		C-CV-B		
		C-MOV-A		
CCS-TRAINS	AND	CCS-TRAIN-B	CCS-NEW-G2	
CCS-NEW-G2	AND	DG-C	DG-D	

Figure 191. The Fault Tree Editor Options Menu.

The options are described as follows:

- o Veto. Veto causes each operation to ask again if the user wishes to modify the data. By default this is turned off. If you are not familiar with the Fault Tree Editor, it may be useful to turn this on when starting the Editor. An important use of this feature is using the Find and Replace with veto on to selectively replace a large number of inputs. Each one would be displayed on the screen, giving us the choice of replacing or not replacing each occurrence. Figure 192 shows the message used when veto on. We positioned at either veto field, and pressed the <Esc> key to see the additional veto help shown in Figure 193.
- o Veto Delete. Veto Delete functions the same as Veto above except it applies only when deleting a gate.
- o Help. Help allows the user to turn off the help menus, as shown in Figure 194. The operations function as before. This is used when an accomplished user does not want the menus displayed. We positioned at the help field and pressed the Esc key to see the additional help shown in Figure 195.

FAULT TREE EDITOR				
Veto				
Option Modify / Skip / Continue to end / End				
Select an option and press <Enter> when done, and <Esc> for help				
Gate Name	Type	Inputs		
CCS-NEW	AND	TANK	C-PUMP-A	CCS-SUPPLY
C-MOV-1-FAILS	OR	C-MOV-A	C-MOV-A	
CCS-SUPPLY	OR	C-MOV-1	TANK	C-PUMP-A
CCS-TOP	OR	C-MOV-1-FAILS	CCS-TRAINS	
CCS-TRAIN-B	OR	CCS-SUPPLY	C-MOV-B	C-PUMP-B
		C-CV-B		
		C-MOV-A		
CCS-TRAINS	AND	CCS-TRAIN-B	CCS-NEW-G2	
CCS-NEW-G2	AND	DG-C	DG-D	

Figure 192. The Veto Message.

<p>When Veto is on you will be prompted to</p> <p><M>odify current gate or input</p> <p><S>kip modification</p> <p><C>ontinue without asking, and</p> <p><E>nd operation.</p>

Figure 193. Help Message for Find and Insert/Replace.

FAULT TREE EDITOR				
Gate Name	Type	Inputs		
CCS-NEW	AND	TANK	C-PUMP-A	CCS-SUPPLY
C-MOV-1-FAILS	OR	C-MOV-A	C-MOV-A	
CCS-SUPPLY	OR	C-MOV-1	TANK	C-PUMP-A
CCS-TOP	OR	C-MOV-1-FAILS	CCS-TRAINS	
CCS-TRAIN-B	OR	CCS-SUPPLY	C-MOV-B	C-PUMP-B
		C-CV-B		
CCS-TRAINS	AND	C-MOV-A	CCS-NEW-G2	
CCS-NEW-G2	AND	CCS-TRAIN-B	DG-D	
		DG-C		

Figure 194. The Help Menus Turned Off.

Hide menu allows the polished user to turn on or off the commands and cursor menus. Esc will always turn the menus back on, and "ON" is the default value.

Figure 195. Help Message for Hide Menu.

Finally, when we were done with the editing of the fault trees, we pressed the Exit command or the "E" key and the screen cleared. We were given three choices, as shown in Figure 196.

Figure 196. Check and save data inquiry.

Either the C or R options will cause the data to be checked using the following rules:

- o Each gate must have a legal gate type chosen from the list in Figure 197.
- o All gates except TRAN must have at least one input.
- o The number of inputs must agree with the gate type. For example a 3/5 gate must have 5 inputs. AND, OR, TAB, NAND, and NOR gates are not limited to the number of inputs except that at least one is required.

AND	3/8
OR	3/9
TAB	4/5
TRAN	4/6
NAND	4/7
NOR	4/8
2/3	4/9
2/4	5/6
2/5	5/7
2/6	5/8
2/7	5/9
2/8	6/7
2/9	6/8
3/4	6/9
3/5	7/8
3/6	7/9
3/7	8/9

Figure 197. Legal gate types.

If an error exists in the data, the check routine will mark the first one encountered by placing the cursor on the bad gate and return to the editing functions. When the data checks successfully, it will be saved (C) or we will be returned to the editing functions (R).

We entered an E to bypass the check and bypass saving the data, leaving it the same as it was when we started.

4.12 MAR-D Data Base Interface

IRRAS 2.0 has the capability to convert most of the information contained in its data base to and from a generic format called the Models And Results Data base format (MAR-D format). This will make it easier to share the information in the IRRAS 2.0 data base with other users and to convert it to and from the formats needed by other Probabilistic Risk Assessment (PRA) tools.

Figure 198. UTILITY Options Menu.

NOTE: There are some naming conventions that data in the MAR-D format must follow. The main convention is that there can be no spaces in the middle of any data element (sequence, fault tree, family) name. To meet this convention, any spaces in a data element name will be replaced with an underscore (_). For example, a family named "demo family" would become "demo_family" when written out to a MAR-D file. The operating system also requires that file names, which are truncated data element names, must not contain spaces. For example, a file from this same "demo family" would have the file name of "demo_fam.fad".

When reading data from a MAR-D file into IRRAS 2.0, the data element names will be as read from the file. In other words, if the data element name read from the file was "demo_family", it would be loaded into the IRRAS 2.0 data base as "demo_family", not "demo family." It is then possible to unload data for the "demo family" family to a MAR-D file and load it back into the IRRAS 2.0 data base as data for the "demo_family" family.

4.12.1 Converting From IRRAS 2.0 Format to MAR-D Format

To convert IRRAS 2.0 data base records into files in the MAR-D format, go into the UTILITY Options menu, which is shown in Figure 198, and select option M - MAR-D Data Base Interface. This option displays the IRRAS / MAR-D Interface menu shown in Figure 199. After selecting the desired family (see Section 3.3.1) or ensuring that the displayed family is the desired one, select option I - IRRAS to MAR-D. This displays the IRRAS to MAR-D conversion options menu shown in Figure 200.

Figure 199. IRRAS-PC IRRAS / MAR-D Interface menu.

Figure 200. IRRAS to MAR-D Conversion Options menu.

Information relating to families, basic events, fault trees, event trees, and sequences can be converted into files in the MAR-D format. The actual process of converting each type of information will be described below. If any files to be written out already exist, the existing files will be overwritten. To prevent the loss of valuable files, make copies of any files you do not want to lose, before converting IRRAS 2.0 data into files in the MAR-D format.

4.12.1.1 Family. Regardless of the selected family, the descriptive information about any family in the IRRAS 2.0 data base can be converted. Select option F - Family to display the IRRAS to MAR-D family conversion options menu shown in Figure 201.

Figure 201. IRRAS to MAR-D Family Conversion Options menu.

To convert the family names and descriptions, select option N, which displays the family name and description conversion menu shown in Figure 202.

I R R A S T o M A R - D C o n v e r s i o n

Text Conversion

Option |C| Exit / Convert Family

NAME	DESCRIPTION
DEMO_FAMILY	Demonstration sample family

NOTE: To convert a Family, enter option <C>, move the cursor to the desired Family, and press <Enter>. To convert all Families, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Families.

Figure 204. IRRAS to MAR-D Family Text Conversion menu.

This menu works just like the name and description menu described above. The text conversion creates a file in the format shown in Figure 205. The name of this file is the first eight characters of the family name with .FAT appended.

```

File Format:
  family name =
  -- text --

where:
  family name   - 16 character family name

General Rule:
  1. File is standard single spaced ASCII text,
     with the start of paragraph indented 5 spaces and
     a blank line between paragraphs.

```

Figure 205. MAR-D Family Text File Format.

4.12.1.2 Basic Event. The basic event information of the selected family can be converted into files in the MAR-D format. Select option B - Basic Event to display the IRRAS to MAR-D basic event conversion options menu shown in Figure 206.

Figure 206. IRRAS to MAR-D Basic Event Conversion Options.

To convert the basic event names and descriptions, select option N, which displays the basic event name and description conversion menu shown in Figure 207.

Figure 207. IRRAS to MAR-D Basic Event Name and Description Conversion Menu.

To convert all of the basic event names and descriptions for the currently selected family to the MAR-D format, select option C and press <Enter>. This conversion creates a file in the format

shown in Figure 208. The name of this file is the first eight characters of the current family name with .BED appended.

```
File Format:
family name =
name,description
... , ...

where:
family name   - 16 character family name
name          - 16 character unique event name
description   - maximum of 60 character alphanumeric
               description
```

Figure 208. MAR-D Event Name and Description File Format.

To convert the basic event probability information, select option P, which displays the basic event probability conversion menu shown in Figure 209.

```
IRRAS To MAR-D Conversion

Probability Conversion

Currently Selected Family —> DEMO_FAMILY

Option |C| Exit / Convert Events

Note: To convert all Events for the Selected Family, enter option <C> and
      press <Enter>. When conversion is complete, or if you do not wish to
      convert the Events, enter option <E> and press <Enter>.
```

Figure 209. IRRAS to MAR-D Basic Event Probability Conversion Menu.

This menu works just like the basic event name and description menu described above. The probability conversion creates a file in the format shown in Figure 210. The name of this file is the first eight characters of the current family name with .BEI appended.

```

File Format:
family name =
name, calc, uncC, uncT, uncV, prob, lamda, tau, init
..., ..., ..., ..., ..., ..., ..., ..., ...

where:
family name - 16 character family name
name        - 16 character unique event name
calc        - 1 character calculation type where:
              1 - Probability
              2 - Lambda * Mission Time
              3 - 1 - Exp(-Lambda * Mission Time)
              4 - Lambda * Min(Mission Time, Tau)
              5 - Operating component with full repair
              6 - Lambda * Tau / 2.0
              7 - 1 + (EXP(-Lambda*Tau)-1.0)/(Lambda*Tau)
              8 - Base Probability * Probability
              9 - Base Probability * Probability
              T - Set to House Event (Failed, Prob=1.0)
              F - Set to House Event (Successful, Prob=0.0)

uncC        - uncertainty correlation
              (Integer - length of 3)
              Events in same class are 100% correlated.
uncT        - 1 character uncertainty distribution type
uncV        - uncertainty value (Floating point)
prob        - probability value (Floating point)
lamda       - basic event failure rate per hour
              (Floating point)
tau         - time to repair in hours (Floating point)
init        - initiating event (Boolean - T/F)

```

Figure 210. MAR-D Probability File Format.

To convert the basic event attributes, select option A, which displays the basic event attribute conversion menu shown in Figure 211.

<div style="border: 1px solid black; display: inline-block; padding: 2px 10px;">I R R A S T o M A R - D C o n v e r s i o n</div>
<p>Attribute Conversion</p> <p>Currently Selected Family —> DEMO_FAMILY</p> <p>Option C Exit / Convert Events</p> <p>Note: To convert all Events for the Selected Family, enter option <C> and press <Enter>. When conversion is complete, or if you do not wish to convert the Events, enter option <E> and press <Enter>.</p>

Figure 211. IRRAS to MAR-D Basic Event Attribute Conversion Menu.

This menu works just like the basic event name and description menu described above. The attribute conversion creates a file with the format shown in Figure 212. The name of this file is the first eight characters of the current family name with .BEA appended.

File Format:	
family name =	
name,altName,type,sys,fail,loc,compID,train,att1,...,att16	
where:	
family name	- 16 character family name
name	- 16 character basic event name
altName	- 16 character alternate basic event name default is value of "name"
type	- 3 character upper case alphanumeric component type
sys	- 3 character upper case alphanumeric system type
fail	- 2 character upper case alphanumeric failure mode
loc	- 3 character upper case alphanumeric component location
compID	- 7 character alphanumeric component ID
train	- 1 character train (Not used in IRRAS)
att1 ... att16	- class attribute susceptibilities, which are 16 values of Y or N (for yes or no) that indicate whether the attribute is applicable for this event.

Figure 212. MAR-D Attribute File Format.

There are other records in the IRRAS 2.0 data base that describe some of the basic event attributes. To convert these records, select option 0, which displays the basic event continuation menu shown in Figure 213.

Figure 213. IRRAS to MAR-D Basic Event Continuation Menu.

To convert the basic event failure mode information, select option F, which displays the basic event failure conversion menu shown in Figure 214.

Figure 214. IRRAS to MAR-D Basic Event Failure Conversion menu.

This menu works just like the basic event name and description menu described above. The failure conversion creates a file in

the format shown in Figure 215. The name of this file is the first eight characters of the current family name with .FMD appended.

```
File Format:
family name =
fail,description
... ,...

where:
family name - 16 character family name
fail        - failure mode identifier
              2 character upper case alphanumeric
              used in "Basic Event Attribute Codes" file
description - failure mode description
              maximum of 60 alphanumeric characters
```

Figure 215. MAR-D failure file format.

To convert the basic event component type information, select option C, which displays the basic event component conversion menu shown in Figure 216.

```
IRRAS To MAR-D Conversion

Component Conversion

Currently Selected Family --> DEMO_FAMILY

Option |C| Exit / Convert Event Attribute Descriptions

Note: To convert all Descriptions for the Selected Family, enter option <C>
and press <Enter>. When conversion is complete, or if you do not wish
to convert the Descriptions, enter option <E> and press <Enter>.
```

Figure 216. IRRAS to MAR-D Basic Event Component Conversion menu.

This menu works just like the basic event name and description menu described above. The component conversion creates a file in the format shown in Figure 217. The name of this file is the first eight characters of the current family name with .CTD appended.

```

File Format:
  family name =
  comp, description
  ... ,...

where:
  family name   - 16 character family name
  comp          - component type identifier
                  3 character upper case alphanumeric
                  used in "Basic Event Attribute Codes" file
  description   - component type description
                  maximum of 60 alphanumeric characters

```

Figure 217. MAR-D Component File Format.

To convert the basic event system type conversion, select option S, which displays the basic event system conversion menu shown in Figure 218.

```

      I R R A S   T o   M A R - D   C o n v e r s i o n

      System Conversion

      Currently Selected Family ----> DEMO_FAMILY

      Option |C| Exit / Convert Event Attribute Descriptions

      Note: To convert all Descriptions for the Selected Family, enter option <C>
            and press <Enter>. When conversion is complete, or if you do not wish
            to convert the Descriptions, enter option <E> and press <Enter>.

```

Figure 218. IRRAS to MAR-D Basic Event System Conversion menu.

This menu works just like the basic event name and description menu described above. The system conversion creates a file in the format shown in Figure 219. The name of this file is the first eight characters of the current family name with .STD appended.

```

File Format:
  family name =
  sys,description
  ... ,...

where:
  family name - 16 character family name
  sys         - component system identifier
               3 character upper case alphanumeric
               used in "Basic Event Attribute Codes" file
  description - component system description
               maximum of 60 alphanumeric characters

```

Figure 219. MAR-D System File Format.

To convert the basic event location information, select option L, which displays the basic event location conversion menu shown in Figure 220.

```

      I R R A S   T o   M A R - D   C o n v e r s i o n

      Location Conversion

      Currently Selected Family —> DEMO_FAMILY

      Option [C] Exit / Convert Event Attribute Descriptions

      Note: To convert all Descriptions for the Selected Family, enter option <C>
            and press <Enter>. When conversion is complete, or if you do not wish
            to convert the Descriptions, enter option <E> and press <Enter>.

```

Figure 220. IRRAS to MAR-D basic event location conversion menu.

This menu works just like the basic event name and description menu described above. The location conversion creates a file in the format shown in Figure 221. The name of this file is the first eight characters of the current family name with .LCD appended.

```

File Format:
family name =
loc,description
... ,...

where:
family name - 16 character family name
loc         - component location identifier
              3 character upper case alphanumeric
              used in "Basic Event Attribute Codes" file
description - component location description
              maximum of 60 alphanumeric characters

```

Figure 221. MAR-D location file format.

To convert the basic event class attributes, select option A, which displays the basic event attribute class conversion menu shown in Figure 222.

```

      I R R A S   T O   M A R - D   C o n v e r s i o n

Attribute Class Conversion

Currently Selected Family —> DEMO_FAMILY

Option [C] Exit / Convert Event Attribute Descriptions

Note: To convert all Descriptions for the Selected Family, enter option <C>
      and press <Enter>. When conversion is complete, or if you do not wish
      to convert the Descriptions, enter option <E> and press <Enter>.

```

Figure 222. IRRAS to MAR-D basic event attribute class conversion menu.

This menu works just like the basic event name and description menu described above. The attribute class conversion creates a file in the format shown in Figure 223. The name of this file is the first eight characters of the current family name with .CAD appended.


```

File Format:
family name =
Attr#,description
... ,...
where:
    family name - 16 character family name
    Attr#       - which attribute number
    description  - maximum of 60 alphanumeric characters
                  (e.g. susceptible to fire or flood)
                  set as applicable for an event in the
                  "Basic Event Attribute Codes" file

```

Figure 223. MAR-D attribute class file format.

4.12.1.3 Fault Tree. The fault tree information of the selected family can be converted into files in the MAR-D format. Select option T - Fault Tree to display the IRRAS to MAR-D fault tree conversion options menu shown in Figure 224.

Figure 224. IRRAS to MAR-D fault tree conversion options menu.

To convert fault tree names and descriptions, select option N, which displays the fault tree name and description conversion menu shown in Figure 225.

I R R A S T o M A R - D C o n v e r s i o n

Name and Description Conversion

Currently Selected Family —> DEMO_FAMILY

Option |C| Exit / Convert Fault Tree

NAME	DESCRIPTION
CCS	Containment Cooling System fault tree
ECS	Emergency Cooling System fault tree
LOSP	Loss of Offsite Power Initiating Event fault tree

Note: To convert a Fault Tree, enter option <C>, move the cursor to the needed Tree, and press <Enter>. To convert all Fault Trees enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Fault Trees.

Figure 225. IRRAS to MAR-D fault tree name and description conversion menu.

This menu lists all the fault trees of the selected family in the IRRAS 2.0 data base. From this menu, select option C, highlight the desired fault tree, and press <Enter>. If all of the fault trees need to be converted, select option C and press <Enter>. This conversion creates a file in the format shown in Figure 226. The name of this file is the first eight characters of the family name with .FTD appended.

File Format:

```
family name =
name, description
... , ...
```

where:

family name	- 16 character family name
name	- 16 character unique fault tree name
description	- maximum of 60 character alpha-numeric description

Figure 226. MAR-D fault tree description file format.

To convert the fault tree graphics data, select option G, which displays the fault tree graphics conversion menu shown in Figure 227.

I R R A S T o M A R - D C o n v e r s i o n

Graphics Conversion

Currently Selected Family —> DEMO_FAMILY

Option |C| Exit / Convert Fault Tree

NAME	DESCRIPTION
CCS	Containment Cooling System fault tree
ECS	Emergency Cooling System fault tree
LOSP	Loss of Offsite Power Initiating Event fault tree

Note: To convert a Fault Tree, enter option <C>, move the cursor to the needed Tree, and press <Enter>. To convert all Fault Trees enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Fault Trees.

Figure 227. IRRAS to MAR-D fault tree graphics conversion menu.

This menu lists all the fault trees of the selected family in the IRRAS 2.0 data base. From this menu, select option C, highlight the desired fault tree, and press <Enter>. If all of the fault trees need to be converted, select option C and press <Enter>. This graphics conversion creates files in the format that the fault tree build module creates. The names of these files are the first eight characters of the fault tree name with .DLS appended.

To convert the fault tree logic information, select option L, which displays the fault tree logic conversion menu shown in Figure 228.

I R R A S T o M A R - D C o n v e r s i o n

Logic Conversion

Currently Selected Family —> DEMO_FAMILY

Option |C| Exit / Convert Fault Tree

NAME	DESCRIPTION
CCS	Containment Cooling System fault tree
ECS	Emergency Cooling System fault tree
LOSP	Loss of Offsite Power Initiating Event fault tree

Note: To convert a Fault Tree, enter option <C>, move the cursor to the needed Tree, and press <Enter>. To convert all Fault Trees enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Fault Trees.

Figure 228. IRRAS to MAR-D fault tree logic conversion menu.

This menu works just like the fault tree graphics menu described above. The fault tree logic conversion creates files in the format shown in Figure 229. The names of these files are the first eight characters of the fault tree name with .FTL appended.

```

File Format:
family name, fault tree name =
gatenam1 gatetype input1 input2 ... inputn
...
gatenamen gatetype input1 input2 ... inputn

where:
family name      - 16 character family name
fault tree name - 16 character fault tree name
gatenam         - maximum 16 character name of the gate
gatetype        - BE      = basic event
                  AND      = logical AND
                  OR       = logical OR
                  TBL      = table of events
                  TRAN     = transfer - 16 character name
                           following is a fault tree name
                  NAND     = logical NOT AND
                  NOR      = logic NOT OR
                  N/M      = N out of M logic gate
                  CONT     = continuation gate
input           - inputs to the gate (basic event or gate
                  names)

General Rules concerning the format of MAR-D fault tree logic
files are:
(1) Each line constitutes a gate definition unless the
    next line has a continuation gate type. In such
    cases, the inputs that follow the continuation
    gate(s) are assigned to the last gate that did not
    have a continuation gate type.
(2) Complimented events are preceded by a slash ("/").
(3) A line starting with an asterisk ("*") is a comment.

```

Figure 229. MAR-D fault tree logic file format.

To convert the fault tree cutsets information, select option C, which displays the fault tree cutsets conversion menu shown in Figure 230.

I R R A S T o M A R - D C o n v e r s i o n

Cutset Conversion

Currently Selected Family —> DEMO_FAMILY

Option |C| Exit / Convert Fault Tree

NAME	DESCRIPTION
CCS	Containment Cooling System fault tree
ECS	Emergency Cooling System fault tree
LOSP	Loss of Offsite Power Initiating Event fault tree

Note: To convert a Fault Tree, enter option <C>, move the cursor to the needed Tree, and press <Enter>. To convert all Fault Trees enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Fault Trees.

Figure 230. IRRAS to MAR-D fault tree cutset conversion menu.

This menu works just like the fault tree graphics menu described above. The fault tree cutsets conversion creates files in the format shown in Figure 231. The names of these files are the first eight characters of the fault tree name with .FTC appended.

```

File Format:
family name, fault tree name =
eventname * eventname +
eventname * eventname * eventname *
eventname +
...      ....      ....
eventname * eventname.

where:
family name      - 16 character family name
fault tree name- 16 character fault tree name
eventname        - basic event names in the cut set.

General Rules:
(1) Events in a cut set are separated by asterisks ("**").
    Spaces are ignored.
(2) Cut sets are separated by a plus ("+" ).
(3) A period (".") denotes the last cutset.
(4) Complimented events are preceded by a slash ("/").
(5) A line starting with an asterisk ("*") is a comment.

```

Figure 231. MAR-D fault tree cutsets file format.

To convert the fault tree attributes, select option A, which displays the fault tree attribute conversion menu shown in Figure 232.

IRRAS To MAR-D Conversion

Attribute Conversion

Currently Selected Family —> DEMO_FAMILY

Option |C| Exit / Convert Fault Tree

NAME	DESCRIPTION
CCS	Containment Cooling System fault tree
ECS	Emergency Cooling System fault tree
LOSP	Loss of Offsite Power Initiating Event fault tree

Note: To convert a Fault Tree, enter option <C>, move the cursor to the needed Tree, and press <Enter>. To convert all Fault Trees enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Fault Trees.

Figure 232. IRRAS to MAR-D fault tree attribute conversion menu.

This menu works just like the fault tree graphics menu described above. The fault tree attribute conversion creates files in the format shown in Figure 233. The name of this file is the first eight characters of the current family name with .FTA appended.


```

File Format:
family name =
name, level, mission, point, probcut, sample, seed, sizecut
..., ..., ..., ..., ..., ..., ..., ...

family name - 16 character family name
name        - 16 character fault tree name
level       - tree level number (Integer - length of 4)
              level of 0 is a top level tree
mission      - mission time (Floating point)
point        - point estimate (Floating point)
probcut      - probability cut off value (Floating point)
sample       - sample size (Integer - length of 4)
seed         - random number seed (Integer - length of 8)
sizecut      - size cut off value (Integer - length of 2)

General Rules:
(1) If a piece of information is not available the comma
    ("") should be included as a placeholder.
(2) Extra blanks between fields are ignored.

```

Figure 233. IRRAS to MAR-D fault tree attribute file format.

To convert the fault tree text information, select option T, which displays the fault tree text conversion menu shown in Figure 234.

I R R A S T o M A R - D C o n v e r s i o n

Text Conversion

Currently Selected Family —> DEMO_FAMILY

Option |C| Exit / Convert Fault Tree

NAME	DESCRIPTION
CCS	Containment Cooling System fault tree
ECS	Emergency Cooling System fault tree
LOSP	Loss of Offsite Power Initiating Event fault tree

Note: To convert a Fault Tree, enter option <C>, move the cursor to the needed Tree, and press <Enter>. To convert all Fault Trees enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Fault Trees.

Figure 234. IRRAS to MAR-D fault tree test conversion menu.

This menu works just like the fault tree graphics menu described above. The fault tree text conversion creates files in the format shown in Figure 235. The name of this file is the first eight characters of the fault tree name with .FTT appended.

File Format:

family name, fault tree name =

-- text --

where:

family name - 16 character family name

fault tree name - 16 character fault tree name

General Rule:

(1) File is standard single spaced ASCII text,
with the start of paragraph indented 5 spaces and
a blank line between paragraphs.

Figure 235. MAR-D fault tree test file format.

4.12.1.4 Event Tree. The event tree information of the selected family can be converted into files in the MAR-D format.

Select option V - Event Tree to display the IRRAS to MAR-D event tree conversion options menu shown in Figure 236.

Figure 236. IRRAS to MAR-D event tree conversion options menu.

To convert event tree names and descriptions, select option N, which displays the event tree name and description conversion menu shown in Figure 237.

I R R A S T o M A R - D C o n v e r s i o n

Name and Description Conversion

Currently Selected Family —> DEMO_FAMILY

Option |C| Exit / Convert Event Tree

NAME	DESCRIPTION
LOSP	Loss of offsite power event tree

Note: To convert an Event Tree, enter option <C>, move the cursor to the needed Tree, and press <Enter>. To convert all Event Trees enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Event Trees.

Figure 237. IRRAS to MAR-D event tree name and description conversion menu.

This menu lists all the event trees of the selected family in the IRRAS data base. From this menu, select option C, highlight the desired event tree, and press <Enter>. If all of the event trees need to be converted, select option C and press <Enter>. This conversion creates a file in the format shown in Figure 238. The name of this file is the first eight characters of the family name with .ETD appended.

```

File Format:
family name =
name,description
... ,...

where:
family name   - 16 character family name
name          - 16 character unique event name
description   - maximum of 60 character alpha-
                  numeric description

```

Figure 238. MAR-D event tree description file format.

To convert the event tree graphics data, select option G, which displays the event tree graphics conversion menu shown in Figure 239.

I R R A S T o M A R - D C o n v e r s i o n

Graphics Conversion

Currently Selected Family —> DEMO_FAMILY

Option |C| Exit / Convert Event Tree

NAME	DESCRIPTION
LOSP	Loss of offsite power event tree

Note: To convert an Event Tree, enter option <C>, move the cursor to the needed Tree, and press <Enter>. To convert all Event Trees enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Event Trees.

Figure 239. IRRAS to MAR-D event tree graphics conversion menu.

This menu lists all the event trees of the selected family in the IRRAS 2.0 data base. From this menu, select option C, highlight the desired event tree, and press <Enter>. If all of the event trees need to be converted, select option C and press <Enter>. This event tree graphics conversion creates files in the same format as the .DLS files. The names of these files are the first eight characters of the event tree name with .ETG appended.

To convert the event tree logic information, select option L, which displays the event tree logic conversion menu shown in Figure 240.

IRRAS To MAR-D Conversion

Logic Conversion

Currently Selected Family —> DEMO_FAMILY

Option |C| Exit / Convert Event Tree

NAME	DESCRIPTION
LOSP	Loss of offsite power event tree

Note: To convert an Event Tree, enter option <C>, move the cursor to the needed Tree, and press <Enter>. To convert all Event Trees enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Event Trees.

Figure 240. IRRAS to MAR-D event tree logic conversion menu.

Currently, the convert option on this menu is not implemented. However, when it is implemented, this menu will work just like the event tree graphics menu described above, with the file names being the first eight characters of the event tree name with .ETL appended.

To convert the event tree attributes, select option A, which displays the event tree attribute conversion menu shown in Figure 241.

I R R A S T o M A R - D C o n v e r s i o n

Attribute Conversion

Currently Selected Family → DEMO_FAMILY

Option |C| Exit / Convert Event Tree

NAME	DESCRIPTION
LOSP	Loss of offsite power event tree

Note: To convert an Event Tree, enter option <C>, move the cursor to the needed Tree, and press <Enter>. To convert all Event Trees enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Event Trees.

Figure 241. IRRAS to MAR-D event tree attribute conversion menu.

This menu works just like the event tree graphics menu described above. The event tree attribute conversion creates files in the format shown in Figure 242. The names of these files are the first eight characters of the event tree name with .ETA appended.

```

File Format:
family name =
name,ie
... ,...

where:
family name   - 16 character family name
name          - 16 character event name
ie            - initiating event name
               16 character basic event name

```

Figure 242. MAR-D event tree attribute file format.

To convert the event tree text information, select option T, which displays the event tree text conversion menu shown in Figure 243.

I R R A S T o M A R - D C o n v e r s i o n

Text Conversion

Currently Selected Family —> DEMO_FAMILY

Option |C| Exit / Convert Event Tree

NAME	DESCRIPTION
LOSP	Loss of offsite power event tree

Note: To convert an Event Tree, enter option <C>, move the cursor to the needed Tree, and press <Enter>. To convert all Event Trees enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Event Trees.

Figure 243. IRRAS to MAR-D event tree text conversion menu.

This menu works just like the event tree graphics menu described above. The event tree text conversion creates files in the format shown in Figure 244. The names of these files are the first eight characters of the event tree name with .ETT appended.

File Format:

family name, event tree name =

-- text --

where:

family name - 16 character family name

event tree name - 16 character event tree name

General Rule:

(1) File is standard single spaced ASCII text,
with the start of paragraph indented 5 spaces and
a blank line between paragraphs.

Figure 244. MAR-D fault tree text file format.

4.12.1.5 Sequence. The sequence information of the selected family can be converted into files in the MAR-D format. Select option S - Sequence to display the IRRAS to MAR-D sequence conversion options menu shown in Figure 245.

Figure 245. IRRAS to MAR-D sequence conversion options menu.

To convert sequence names and descriptions, select option N, which displays the sequence name and description conversion menu shown in Figure 246.

I R R A S T o M A R - D C o n v e r s i o n

Name and Description Conversion

Currently Selected Family —> DEMO_FAMILY

Option |C| Exit / Convert Sequence

NAME	DESCRIPTION
1-LOSP-ECS	Loss of Offsite Power with success of ECS, no core damage.
2-LOSP-ECS-CCS	Loss of Offsite Power, failed ECS, successful CCS, sm. rel
3-LOSP-ECS-CCS	Loss of Offsite Power, failed ECS, failed CCS, lg. release

Note: To convert a Sequence, enter option <C>, move the cursor to the desired Sequence, and press <Enter>. To convert all Sequences, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Sequences.

Figure 246. IRRAS to MAR-D sequence name and description conversion menu.

This menu lists all the sequences of the selected family in the IRRAS data base. From this menu, select option C, highlight the desired sequence, and press <Enter>. If all of the sequences need to be converted, select option C and press <Enter>. This conversion creates a file in the format shown in Figure 247. The name of this file is the first eight characters of the family name with .SQD appended.

File Format:

```
family name =
name,description
... ,...
```

where:

family name	- 16 character family name
name	- maximum of 16 character unique sequence name
description	- maximum of 60 character description.

Figure 247. MAR-D sequence description file format.

To convert the sequence logic information, select option L, which displays the sequence logic conversion menu shown in Figure 248.

I R R A S T o M A R - D C o n v e r s i o n

Logic Conversion

Currently Selected Family —> DEMO_FAMILY

Option |C| Exit / Convert Sequence

NAME	DESCRIPTION
1-LOSP-ECS	Loss of Offsite Power with success of ECS, no core damage.
2-LOSP-ECS-CCS	Loss of Offsite Power, failed ECS, successful CCS, sm. rel
3-LOSP-ECS-CCS	Loss of Offsite Power, failed ECS, failed CCS, lg. release

Note: To convert a Sequence, enter option <C>, move the cursor to the desired Sequence, and press <Enter>. To convert all Sequences, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Sequences.

Figure 248. IRRAS to MAR-D sequence logic conversion menu.

Currently, the convert option on this menu is not implemented. However, when it is implemented, this menu will work just like the sequence name and description menu described above, with the file names being the first eight characters of the event tree name with .SQL appended.

To convert the sequence cutsets information, select option C, which displays the sequence cutsets conversion menu shown in Figure 249.

I R R A S T o M A R - D C o n v e r s i o n

Cutset Conversion

Currently Selected Family —> DEMO_FAMILY

Option |C| Exit / Convert Sequence

NAME	DESCRIPTION
1-LOSP-ECS	Loss of Offsite Power with success of ECS, no core damage.
2-LOSP-ECS-CCS	Loss of Offsite Power, failed ECS, successful CCS, sm. rel
3-LOSP-ECS-CCS	Loss of Offsite Power, failed ECS, failed CCS, lg. release

Note: To convert a Sequence, enter option <C>, move the cursor to the desired Sequence, and press <Enter>. To convert all Sequences, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Sequences.

Figure 249. IRRAS to MAR-D sequence cutsets conversion menu.

This menu works just like the sequence name and description menu described above. The sequence cutsets conversion creates files in the format shown in Figure 250. The names of these files are the first eight characters of the sequence name with .SQC appended.

```

File Format:
  family name, sequence name =
  eventname * eventname +
  eventname * eventname * eventname *
  eventname +
  ...      ....      ....
  eventname * eventname.

where:
  family name   - 16 character family name
  sequence name - 16 character sequence name
  eventname     - basic event names in the cut set.

General Rules:
(1) Events in a cut set are separated by asterisks ("*").
    Spaces are ignored.
(2) Cut sets are separated by a plus ("+").
(3) A period (".") denotes the last cutset.
(4) Complimented events are preceded by a slash ("/").
(5) A line starting with an asterisk ("*") is a comment.

```

Figure 250. MAR-D sequence cutsets file format.

To convert the sequence attributes, select option A, which displays the sequence attribute conversion menu shown in Figure 251.

I R R A S T o M A R - D C o n v e r s i o n

Attribute Conversion

Currently Selected Family —> DEMO_FAMILY

Option |C| Exit / Convert Sequence

NAME	DESCRIPTION
1-LOSP-ECS	Loss of Offsite Power with success of ECS, no core damage.
2-LOSP-ECS-CCS	Loss of Offsite Power, failed ECS, successful CCS, sm. rel
3-LOSP-ECS-CCS	Loss of Offsite Power, failed ECS, failed CCS, lg. release

Note: To convert a Sequence, enter option <C>, move the cursor to the desired Sequence, and press <Enter>. To convert all Sequences, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Sequences.

Figure 251. IRRAS to MAR-D sequence attribute conversion menu.

This menu works just like the sequence name and description menu described above. The sequence attribute conversion creates a file in the format shown in Figure 252. The name of this file is the first eight characters of the family name with .SQA appended.

```

File Format:
family name =
name, et, PDS, point, mission, probcut, sample, seed, size
... , ... , ... , ... , ... , ... , ... , ... , ...

where:
family name - 16 character family name
name        - 16 character sequence name
et          - 16 character event tree name that this
              sequence belongs to.
PDS         - 16 character Plant Damage State name
point       - point estimate (Floating point)
mission     - mission time in hours (Floating point)
probcut     - probability cut off value (Floating point)
sample      - sample size (Integer - length of 4)
seed        - random number seed (Integer - length of 8)
size        - size cut off value (Integer - length of 2)

General Rules:
(1) If a field is unknown leave the comma as a placeholder.

```

Figure 252. MAR-D sequence attribute file format.

To convert the sequence text information, select option T, which displays the sequence text conversion menu shown in Figure 253.

I R R A S T o M A R - D C o n v e r s i o n

Text Conversion

Currently Selected Family —> DEMO_FAMILY

Option |C| Exit / Convert Sequence

NAME	DESCRIPTION
1-LOSP-ECS	Loss of Offsite Power with success of ECS, no core damage.
2-LOSP-ECS-CCS	Loss of Offsite Power, failed ECS, successful CCS, sm. rel
3-LOSP-ECS-CCS	Loss of Offsite Power, failed ECS, failed CCS, lg. release

Note: To convert a Sequence, enter option <C>, move the cursor to the desired Sequence, and press <Enter>. To convert all Sequences, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Sequences.

Figure 253. IRRAS to MAR-D sequence text conversion menu.

This menu works just like the sequence name and description menu described above. The sequence text conversion creates files in the format shown in Figure 254. The names of these files are the first eight characters of the sequence name with .SQT appended.

```

File Format:
  family name, sequence name =
    -- text --

where:
  family name    - 16 character family name
  sequence name  - 16 character sequence name

General Rule:
  (1) File is standard single spaced ASCII text,
      with the start of paragraph indented 5 spaces and
      a blank line between paragraphs.

```

Figure 254. MAR-D sequence attribute file format.

Figure 255. UTILITY options menu.

4.12.2 Converting from MAR-D format to IRRAS format

To convert data from a file in MAR-D format into the IRRAS data base, go into the UTILITY Options menu, which is shown in Figure 255, and select option M - MAR-D Data Base Interface. This option displays the IRRAS-PC IRRAS / MAR-D Interface menu shown in Figure 256. After selecting the correct family (see Section 3.3.1) or ensuring that the displayed family is the desired one, select option M - MAR-D to IRRAS. This displays the MAR-D to IRRAS conversion options menu shown in Figure 257.

Figure 256. IRRAS-PC IRRAS / MAR-D Interface menu.

Figure 257. MAR-D to IRRAS conversion options menu.

Information relating to families, basic events, fault trees, event trees, and sequences can be converted into the IRRAS data base from files in the MAR-D format. The actual process of converting each type of information will be described below. If any data already exists in the data base, it will be replaced by

the data from the MAR-D files. To prevent the loss of valuable data, make copies of the data base before converting MAR-D data into the IRRAS 2.0 data base.

4.12.2.1 Family. Regardless of the selected family, the information about any family in a MAR-D data file can be converted. Select option F - Family to display the MAR-D to IRRAS family conversion options menu shown in Figure 258.

Figure 258. MAR-D to IRRAS family conversion options menu.

To convert the family names and descriptions, select option N, which displays the family name and description conversion menu shown in Figure 259.

Convert From MAR-D To IRRAS

Name and Description Conversion

Currently Selected Family → DEMO_FAMILY

Option |C| Exit / Convert Family

DEMO_FAM

Note: To convert an Family, enter option <C>, move the cursor to the desired Family, and press <Enter>. To convert all Families, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Families.

Figure 259. MAR-D to IRRAS family name and description conversion menu.

This menu lists all the .FAD files in the current directory. From this menu, select option C, highlight the desired file name, and press <Enter>. If all of the files need to be converted, select option C and press <Enter>. The data will be read from the files and loaded into the IRRAS 2.0 data base. The format for these files is shown in Figure 203. If any families already exist in the data base, their old information will be overwritten.

To convert the family text information, select option T, which displays the family text conversion menu shown in Figure 260.

Convert From MAR-D To IRRAS

Text Conversion

Currently Selected Family —> DEMO_FAMILY

Option |C| Exit / Convert Family

DEMO_FAM

Note: To convert an Family, enter option <C>, move the cursor to the desired Family, and press <Enter>. To convert all Families, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Families.

Figure 260. MAR-D to IRRAS family text conversion menu.

This menu lists all the .FAT files in the current directory. From this menu, select option C, highlight the desired file name, and press <Enter>. If all of the files need to be converted, select option C and press <Enter>. The data will be read from the files and loaded into the IRRAS 2.0 data base. The format for these files is shown in Figure 205. If any families already exist in the data base, their old information will be overwritten.

4.12.2.2 Basic Event. The basic event information stored in MAR-D format can be loaded into the IRRAS data base and associated with the currently selected family. Select option B - Basic Event to display the MAR-D to IRRAS basic event conversion options menu shown in Figure 261.

Figure 261. MAR-D to IRRAS basic event conversion options menu.

To convert the basic event names and descriptions, select option N, which displays the basic event name and description conversion menu shown in Figure 262.

Convert From MAR-D To IRRAS

Name and Description Conversion

Currently Selected Family → DEMO_FAMILY

Option |C| Exit / Convert Basic Event

DEMO_FAM

Note: To convert a Set of Events, enter option <C>, move the cursor to the Set Name, and press <Enter>. To convert all Event Sets, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Event Sets.

Figure 262. MAR-D to IRRAS basic event description conversion menu.

This menu lists all the .BED files in the current directory. From this menu, select option C, highlight the desired file name, and press <Enter>. If all of the files need to be converted, select option C and press <Enter>. The data will be read from the files and loaded into the IRRAS 2.0 data base. The format for these files is shown in Figure 208. Regardless of the family name in the MAR-D file, the basic event description can be associated with the currently selected family. If any basic events already exist in the data base, their old information will be overwritten.

To convert the basic event probability information, select option P, which displays the basic event probability conversion menu shown in Figure 263.

Convert From MAR-D To IRRAS

Probability Conversion

Currently Selected Family → DEMO_FAMILY

Option |C| Exit / Convert Basic Event

DEMO_FAM

Note: To convert a Set of Events, enter option <C>, move the cursor to the Set Name, and press <Enter>. To convert all Event Sets, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Event Sets.

Figure 263. MAR-D to IRRAS basic event probability conversion menu.

This menu lists all the .BEI files in the current directory and works like the name and description menu described above. The format for these files is shown in Figure 210. Regardless of the family name in the MAR-D file, the basic event probability information can be associated with the currently selected family. If any basic events already exist in the data base, their old information will be overwritten.

To convert the basic event attributes, select option A, which displays the basic event attribute conversion menu shown in Figure 264.

Convert From MAR-D To IRRAS

Attribute Conversion

Currently Selected Family → DEMO_FAMILY

Option |C| Exit / Convert Basic Event

DEMO_FAM

Note: To convert a Set of Events, enter option <C>, move the cursor to the Set Name, and press <Enter>. To convert all Event Sets, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Event Sets.

Figure 264. MAR-D to IRRAS basic event attribute conversion menu.

This menu lists all the .BEA files in the current directory and works like the name and description menu described above. The format for these files is shown in Figure 212. Regardless of the family name in the MAR-D file, the basic event attributes can be associated with the currently selected family. If any basic events already exist in the data base, their old information will be overwritten.

There are other records, which describe the basic event attributes, that can be loaded into the IRRAS 2.0 data base. To convert these records, select option 0, which displays the basic event continuation menu shown in Figure 265.

Figure 265. MAR-D to IRRAS basic event continuation menu.

To convert the basic event failure mode information, select option F, which displays the basic event failure conversion menu shown in Figure 266.

Convert From MAR-D To IRRAS

Failure Conversion

Currently Selected Family → DEMO_FAMILY

Option |C| Exit / Convert Event Attribute Descriptions

DEMO_FAM

Note: To convert a Description, enter option <C>, move the cursor to the Description, and press <Enter>. To convert all Descriptions, enter option <C>, and press <Enter>. Use <PgUp> and <PgDn> to display more Descriptions.

Figure 266. MAR-D to IRRAS basic event failure conversion menu.

This menu lists all the .FMD files in the current directory and works like the name and description menu described above. The format for these files is shown in Figure 215. Regardless of the family name in the MAR-D file, the basic event failure information can be associated with the currently selected family. If any failure modes already exist in the data base, their old information will be overwritten.

To convert the basic event component type information, select option C, which displays the basic event component conversion menu shown in Figure 267.

Convert From MAR-D To IRRAS

Component Conversion

Currently Selected Family → DEMO_FAMILY

Option |C| Exit / Convert Event Attribute Descriptions

DEMO_FAM

Note: To convert a Description, enter option <C>, move the cursor to the Description, and press <Enter>. To convert all Descriptions, enter option <C>, and press <Enter>. Use <PgUp> and <PgDn> to display more Descriptions.

Figure 267. MAR-D to IRRAS basic event component conversion menu.

This menu lists all the .CTD files in the current directory and works like the name and description menu described above. The format for these files is shown in Figure 217. Regardless of the family name in the MAR-D file, the basic event component information can be associated with the currently selected family. If any component types already exist in the data base, their old information will be overwritten.

To convert the basic event system type information, select option S, which displays the basic event system conversion menu shown in Figure 268.

Convert From MAR-D To IRRAS

System Conversion

Currently Selected Family —> DEMO_FAMILY

Option [C] Exit / Convert Event Attribute Descriptions

DEMO_FAM

Note: To convert a Description, enter option <C>, move the cursor to the Description, and press <Enter>. To convert all Descriptions, enter option <C>, and press <Enter>. Use <PgUp> and <PgDn> to display more Descriptions.

Figure 268. MAR-D to IRRAS basic event system conversion menu.

This menu lists all the .STD files in the current directory and works like the name and description menu described above. The format for these files is shown in Figure 219. Regardless of the family name in the MAR-D file, the basic event system information can be associated with the currently selected family. If any system types already exist in the data base, their old information will be overwritten.

To convert the basic event location information, select option L, which displays the basic event location conversion menu shown in Figure 269.

Convert From MAR-D To IRRAS

Location Conversion

Currently Selected Family → DEMO_FAMILY

Option |C| Exit / Convert Event Attribute Descriptions

DEMO_FAM

Note: To convert a Description, enter option <C>, move the cursor to the Description, and press <Enter>. To convert all Descriptions, enter option <C>, and press <Enter>. Use <PgUp> and <PgDn> to display more Descriptions.

Figure 269. MAR-D to IRRAS basic event location conversion menu.

This menu lists all the .LCD files in the current directory and works like the name and description menu described above. The format for these files is shown in Figure 221. Regardless of the family name in the MAR-D file, the basic event location information can be associated with the currently selected family. If any locations already exist in the data base, their old information will be overwritten.

To convert the basic event class attributes, select option A, which displays the basic event attribute class conversion menu shown in Figure 270.

Convert From MAR-D To IRRAS

Attribute Class Conversion

Currently Selected Family —> DEMO_FAMILY

Option |C| Exit / Convert Event Attribute Descriptions

DEMO_FAM

Note: To convert a Description, enter option <C>, move the cursor to the Description, and press <Enter>. To convert all Descriptions, enter option <C>, and press <Enter>. Use <PgUp> and <PgDn> to display more Descriptions.

Figure 270. MAR-D to IRRAS basic event attribute class conversion menu.

This menu lists all the .CAD files in the current directory and works like the name and description menu described above. The format for these files is shown in Figure 223. Regardless of the family name in the MAR-D file, the basic event attribute class information can be associated with the currently selected family. If any attribute classes already exist in the data base, their old information will be overwritten.

4.12.2.3 Fault Tree. The fault tree information stored in MAR-D format can be loaded into the IRRAS 2.0 data base. Select option T - Fault Tree to display the MAR-D to IRRAS fault tree conversion options menu shown in Figure 271.

Figure 271. MAR-D to IRRAS fault tree conversion options menu.

To convert fault tree names and descriptions, select option N, which displays the fault tree name and description conversion menu shown in Figure 272.

Convert From MAR-D To IRRAS

Name and Description Conversion

Currently Selected Family → DEMO_FAMILY

Option |C| Exit / Convert Fault Tree

DEMO_FAM

Note: To convert a Fault Tree, enter option <C>, move the cursor to the Fault Tree, and press <Enter>. To convert all Fault Trees, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Fault Trees.

Figure 272. MAR-D to IRRAS fault tree name and description menu.

This menu lists all the .FTD files in the current directory. From this menu, select option C, highlight the desired file name, and press <Enter>. If all of the files need to be converted, select option C and press <Enter>. The data will be read from the files and loaded into the IRRAS 2.0 data base. The format for these files is shown in Figure 226. Regardless of the family name in the MAR-D file, the fault tree description can be associated with the currently selected family. If any fault trees already exist in the data base, their old information will be overwritten.

To convert the fault tree graphics data, select option G, which displays the fault tree graphics conversion menu shown in Figure 273.

Convert From MAR-D To IRRAS		
Graphics Conversion		
Currently Selected Family → DEMO_FAMILY		
Option C Exit / Convert Fault Tree		
CCS	ECS	LOSP
<p>Note: To convert a Fault Tree, enter option <C>, move the cursor to the Fault Tree, and press <Enter>. To convert all Fault Trees, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Fault Trees.</p>		

Figure 273. MAR-D to IRRAS fault tree graphics conversion menu.

This menu lists all the .DLS files in the current directory. From this menu, select option C, highlight the desired file name, and press <Enter>. If all of the files need to be converted, select option C and press <Enter>. The data will be read from the files and loaded into the IRRAS 2.0 data base. If any fault trees already exist in the data base, their old information will be overwritten.

To convert the fault tree logic information, select option L, which displays the fault tree logic conversion menu shown in Figure 274.

Convert From MAR-D To IRRAS

Logic Conversion

Currently Selected Family → DEMO_FAMILY

Option |C| Exit / Convert Fault Tree

CCS
ECS
LOSP

Note: To convert a Fault Tree, enter option <C>, move the cursor to the Fault Tree, and press <Enter>. To convert all Fault Trees, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Fault Trees.

Figure 274. MAR-D to IRRAS fault tree logic conversion menu.

This menu works just like the fault tree graphics menu described above, except that it lists all the .FTL files in the current directory. The format for these files is shown in Figure 229.

Regardless of the family name in the MAR-D file, the fault tree logic information can be associated with the currently selected family. If any fault trees already exist in the data base, their old information will be overwritten.

To convert the fault tree cutsets information, select option C, which displays the fault tree cutsets conversion menu shown in Figure 275.

Convert From MAR-D To IRRAS

Cutset Conversion

Currently Selected Family → DEMO_FAMILY

Option |C| Exit / Convert Fault Tree

CCS
ECS
LOSP

Note: To convert a Fault Tree, enter option <C>, move the cursor to the Fault Tree, and press <Enter>. To convert all Fault Trees, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Fault Trees.

Figure 275. MAR-D to IRRAS fault tree cutsets conversion menu.

This menu works just like the fault tree graphics menu described above, except that it lists all the .FTC files in the current directory. The format for these files is shown in Figure 231. Regardless of the family name in the MAR-D file, the fault tree cutsets information can be associated with the currently selected family. If any fault trees already exist in the data base, their old information will be overwritten.

To convert the fault tree attributes, select option A, which displays the fault tree attribute conversion menu shown in Figure 276.

Convert From MAR-D To IRRAS

Attribute Conversion

Currently Selected Family → DEMO_FAMILY

Option |C| Exit / Convert Fault Tree

DEMO_FAM

Note: To convert a Fault Tree, enter option <C>, move the cursor to the Fault Tree, and press <Enter>. To convert all Fault Trees, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Fault Trees.

Figure 276. MAR-D to IRRAS fault tree attribute conversion menu.

This menu works just like the fault tree graphics menu described above, except that it lists all the .FTA files in the current directory. The format for these files is shown in Figure 233. Regardless of the family name in the MAR-D file, the fault tree attributes can be associated with the currently selected family. If any fault trees already exist in the data base, their old information will be overwritten.

To convert the fault tree text information, select option T, which displays the fault tree text conversion menu shown in Figure 277.

Convert From MAR-D To IRRAS

Text Conversion

Currently Selected Family → DEMO_FAMILY

Option |C| Exit / Convert Fault Tree

CCS ECS LOSP

Note: To convert a Fault Tree, enter option <C>, move the cursor to the Fault Tree, and press <Enter>. To convert all Fault Trees, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Fault Trees.

Figure 277. MAR-D to IRRAS fault tree text conversion menu.

This menu works just like the fault tree graphics menu described above, except that it lists all the .FTT files in the current directory. The format for these files is shown in Figure 235. Regardless of the family name in the MAR-D file, the fault tree text can be associated with the currently selected family. If any fault trees already exist in the data base, their old information will be overwritten.

4.12.2.4 Event Tree. The event tree information stored in MAR-D format can be loaded into the IRRAS data base. Select option V - Event Tree to display the MAR-D to IRRAS event tree conversion options menu shown in Figure 278.

Figure 278. MAR-D to IRRAS event tree conversion options menu.

To convert event tree names and descriptions, select option N, which displays the event tree name and description conversion menu shown in Figure 279.

Convert From MAR-D To IRRAS

Name and Description Conversion

Currently Selected Family → DEMO_FAMILY

Option |C| Exit / Convert Event Tree

DEMO_FAM

Note: To convert an Event Tree, enter option <C>, move the cursor to the Event Tree, and press <Enter>. To convert all Event Trees, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Event Trees.

Figure 279. MAR-D to IRRAS event tree name and description conversion menu.

This menu lists all the .ETD files in the current directory. From this menu, select option C, highlight the desired file name, and press <Enter>. If all of the files need to be converted, select option C and press <Enter>. The data will be read from the files and loaded into the IRRAS 2.0 data base. The format for these files is shown in Figure 238. Regardless of the family name in the MAR-D file, the event tree description can be associated with the currently selected family. If any event trees already exist in the data base, their old information will be overwritten.

To convert the event tree graphics data, select option G, which displays the event tree graphics conversion menu shown in Figure 280.

Convert From MAR-D To IRRAS

Graphics Conversion

Currently Selected Family —> DEMO_FAMILY

Option |C| Exit / Convert Event Tree

LOSP

Note: To convert an Event Tree, enter option <C>, move the cursor to the Event Tree, and press <Enter>. To convert all Event Trees, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Event Trees.

Figure 280. MAR-D to IRRAS event tree graphics conversion menu.

This menu lists all the .ETG files in the current directory. From this menu, select option C, highlight the desired file name, and press <Enter>. If all of the files need to be converted, select option C and press <Enter>. The data will be read from the files and loaded into the IRRAS data base. If any event trees already exist in the data base, their old information will be overwritten.

To convert the event tree logic information, select option L, which displays the event tree logic conversion menu shown in Figure 281.

Convert From MAR-D To IRRAS

Logic Conversion

Currently Selected Family → DEMO_FAMILY

Option |C| Exit / Convert Event Tree

LOSP

Note: To convert an Event Tree, enter option <C>, move the cursor to the Event Tree, and press <Enter>. To convert all Event Trees, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Event Trees.

Figure 281. MAR-D to IRRAS event tree logic conversion menu.

This menu is not currently implemented, but when implemented it will work just like the event tree graphics menu described above, except that it lists all the .ETL files in the current directory. Regardless of the family name in the MAR-D file, the event tree logic information can be associated with the currently selected family. If any event trees already exist in the data base, their old information will be overwritten.

To convert the event tree attributes, select option A, which displays the event tree attribute conversion menu shown in Figure 282.

Convert From MAR-D To IRRAS

Attribute Conversion

Currently Selected Family ==> DEMO_FAMILY

Option |C| Exit / Convert Event Tree

DEMO_FAM

Note: To convert an Event Tree, enter option <C>, move the cursor to the Event Tree, and press <Enter>. To convert all Event Trees, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Event Trees.

Figure 282. MAR-D to IRRAS event tree attribute conversion menu.

This menu works just like the event tree graphics menu described above, except that it lists all the .ETA files in the current directory. The format for these files is shown in Figure 242. Regardless of the family name in the MAR-D file, the event tree attributes can be associated with the currently selected family. If any event trees already exist in the data base, their old information will be overwritten.

To convert the event tree text information, select option T, which displays the event tree text conversion menu shown in Figure 283.

Convert From MAR-D To IRRAS

Text Conversion

Currently Selected Family → DEMO_FAMILY

Option |C| Exit / Convert Event Tree

LOSP

Note: To convert an Event Tree, enter option <C>, move the cursor to the Event Tree, and press <Enter>. To convert all Event Trees, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Event Trees.

Figure 283. MAR-D to IRRAS event tree text conversion menu.

This menu works just like the event tree graphics menu described above, except that it lists all the .ETT files in the current directory. The format for these files is shown in Figure 244.

Regardless of the family name in the MAR-D file, the event tree text can be associated with the currently selected family. If any event trees already exist in the data base, their old information will be overwritten.

4.12.2.5 Sequence. The sequence information stored in MAR-D format can be loaded into the IRRAS data base. Select option S - Sequence to display the MAR-D to IRRAS sequence conversion options menu shown in Figure 284.

Figure 284. MAR-D to IRRAS sequence conversion options menu.

To convert sequence names and descriptions, select option N, which displays the sequence name and description conversion menu shown in Figure 285.

Convert From MAR-D To IRRAS

Name and Description Conversion

Currently Selected Family —> DEMO_FAMILY

Option |C| Exit / Convert Sequence

DEMO_FAM

Note: To convert a Sequence, enter option <C>, move the cursor to the desired Sequence, and press <Enter>. To convert all Sequences, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Sequences.

Figure 285. MAR-D to IRRAS sequence name and description conversion menu.

This menu lists all the .SQD files in the current directory. From this menu, select option C, highlight the desired file name, and press <Enter>. If all of the files need to be converted, select option C and press <Enter>. The data will be read from the files and loaded into the IRRAS 2.0 data base. The format for these files is shown in Figure 247. Regardless of the family name in the MAR-D file, the sequence description can be associated with the currently selected family. If any sequences already exist in the data base, their old information will be overwritten.

To convert the sequence logic information, select option L, which displays the sequence logic conversion menu shown in Figure 286.

Convert From MAR-D To IRRAS

Logic Conversion

Currently Selected Family → DEMO_FAMILY

Option |C| Exit / Convert Sequence

1-LOSP-E 2-LOSP-E 3-LOSP-E

Note: To convert a Sequence, enter option <C>, move the cursor to the desired Sequence, and press <Enter>. To convert all Sequences, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Sequences.

Figure 286. MAR-D to IRRAS sequence logic conversion menu.

This menu is not currently implemented, but when implemented it will work just like the event tree graphics menu described above, except that it lists all the .SQL files in the current directory. Regardless of the family name in the MAR-D file, the sequence logic information can be associated with the currently selected family. If any sequences already exist in the data base, their old information will be overwritten.

To convert the sequence cutsets information, select option C, which displays the sequence cutsets conversion menu shown in Figure 287.

Convert From MAR-D To IRRAS

Cutset Conversion

Currently Selected Family → DEMO_FAMILY

Option |C| Exit / Convert Sequence

1-LOSP-E 2-LOSP-E 3-LOSP-E

Note: To convert a Sequence, enter option <C>, move the cursor to the desired Sequence, and press <Enter>. To convert all Sequences, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Sequences.

Figure 287. MAR-D to IRRAS sequence cutsets conversion menu.

This menu works just like the sequence name and description menu described above, except that it lists all the .SQC files in the current directory. The format for these files is shown in Figure 250. Regardless of the family name in the MAR-D file, the sequence cutset information can be associated with the currently selected family. If any sequences already exist in the data base, their old information will be overwritten.

To convert the sequence attributes, select option A, which displays the sequence attribute conversion menu shown in Figure 288.

Convert From MAR-D To IRRAS

Attribute Conversion

Currently Selected Family —> DEMO_FAMILY

Option |C| Exit / Convert Sequence

DEMO_FAM

Note: To convert a Sequence, enter option <C>, move the cursor to the desired Sequence, and press <Enter>. To convert all Sequences, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Sequences.

Figure 288. MAR-D to IRRAS sequence attribute conversion menu.

This menu works just like the sequence name and description menu described above, except that it lists all the .SQA files in the current directory. The format for these files is shown in Figure 252. Regardless of the family name in the MAR-D file, the sequence attributes can be associated with the currently selected family. If any sequences already exist in the data base, their old information will be overwritten.

To convert the sequence text information, select option T, which displays the sequence text conversion menu shown in Figure 289.

Convert From MAR-D To IRRAS

Text Conversion

Currently Selected Family ==> DEMO_FAMILY

Option [C] Exit / Convert Sequence

1-LOSP-E 2-LOSP-E 3-LOSP-E

Note: To convert a Sequence, enter option <C>, move the cursor to the desired Sequence, and press <Enter>. To convert all Sequences, enter option <C> and press <Enter>. Use <PgUp> and <PgDn> to display more Sequences.

Figure 289. MAR-D to IRRAS sequence text conversion menu.

This menu works just like the sequence name and description menu described above, except that it lists all the .SQT files in the current directory. The format for these files is shown in Figure 254. Regardless of the family name in the MAR-D file, the sequence text can be associated with the currently selected family. If any sequences already exist in the data base, their old information will be overwritten.

5. FOR THE CURIOUS - A DISCUSSION OF ALGORITHMS

5.1 How Are Minimal Cut Sets Found?

This section describes the method used to progress through the fault tree to generate the cut sets for the fault tree and then make the comparisons necessary to eliminate supersets and reduce those cut sets to the minimal cut sets.

5.1.1 Coalescing of Gates

A gate which has another gate input to it that is the same type may be combined into one gate without changing the logic of the fault tree. For example, Figure 290 shows a single OR gate that is equivalent to a combination of an OR gate being input to another OR gate. This holds true for AND gates as well. Coalescing gates reduces the number of gates which need to be considered in the fault tree reduction. IRRAS 2.0 passes through the fault tree combining gates as described above. The logic of the fault tree is not changed by this conversion, and the processing time is usually reduced.

Figure 290. Coalescing of OR gates.

5.1.2 Identifying Independent Subtrees

The process of identifying independent subtrees in IRRAS 2.0 is a very simple method that determines those portions of the fault tree that appear only once. Independent subtrees can be processed much later in the reduction process and thus tend to reduce the processing time.

5.1.3 Treatment of Special Gates and Events

The logic processed for the generation of cut sets must be reduced to terms involving AND gates, OR gates, and basic events only. Therefore, interpretation and conversion of other types of gates and events must be performed.

5.1.3.1 Treatment of N/M OR Gates. The N/M OR gate generates all the combinations of failures involving N different basic events out of a possible total of M events. This logic is converted to an OR gate with a number of AND gates feeding into it. Each AND gate has N basic events feeding into it. The number of AND gates under the OR gates is determined by the total number of combinations of N failures out of a population of M events. The equation for this number of combinations is

$$M!/[N! * (M - N)!] \quad (5)$$

Thus, for a 2 out of 3 gate, there are 3 unique combinations of 2 failures. This generates 3 AND gates under the OR gate. This example is illustrated in Figure 291.

Figure 291. Conversion of a 2-out-of-3 gate.

5.1.3.2 Treatment of Inhibit Gates. An INHIBIT gate is used to specify some qualifying condition that must be satisfied before the input can produce the output. For example, consider the following situation. If a frozen pipeline is an event of interest, such an event can only occur when the temperature T is less than 32°F . In this case, the output event would be "frozen pipeline," the input event would be "existence of low temperature," and the conditional input would be " $T < 32^{\circ}\text{F}$." Using the INHIBIT gate, this looks like Figure 292.

Figure 292. Example use of an INHIBIT gate.

The INHIBIT gate is actually a special case of the AND gate. Thus the INHIBIT gate is converted to an AND gate with the input events being the original input event and the conditioning event.

5.1.3.3 Treatment of Undeveloped Events and Transfers. Undeveloped events and transfers are used to indicate where further logic model development may be required. Until the logic is expanded, the fault tree can be processed with the undeveloped events and transfers acting as basic events. Data can be assigned and processed as with regular basic events.

5.1.3.4 Treatment of Transfer (TRANSIN) Gates. A transfer gate links the logic of one fault tree to another fault tree. Whenever a transfer gate is encountered in the logic of a fault tree, that line in the logic is replaced by the logic of the fault tree called out in the transfer. If no transfer exists, then the Fault Tree Cut Sets are generated as if the transfer were a basic

event. When the cut sets are displayed, the transfer gates which do not exist are displayed as undeveloped gates.

5.1.3.5 Treatment of HOUSE Events. A HOUSE event is used in a fault tree to designate an event that is always going to happen (true) or never going to happen (false). The presence of a HOUSE event will cause the fault tree to be pruned according to the logic dictated by the HOUSE event.

For a HOUSE event, the following is known:

1. A HOUSE event never appears in the cut sets for a fault tree.
2. A true HOUSE event always satisfies an OR gate, regardless of the other inputs to the gate.
3. A true HOUSE event is understood to be included in the output of an AND gate and need not appear.
4. A false HOUSE event can never satisfy an OR gate.
5. A false HOUSE event as an input to an AND gate prevents the AND gate from having a valid output, since all the inputs to the AND gate are never present.

From these considerations, the following actions are taken by IRRAS 2.0 whenever a HOUSE event is encountered in the processing of a fault tree.

1. If a true HOUSE event is an input to an OR gate, the OR gate is replaced by the HOUSE event.
2. If a true HOUSE event is an input to an AND gate, the HOUSE event is pruned from the tree.
3. If a false HOUSE event is an input to an OR gate, the HOUSE event is pruned from the tree.
4. If a false HOUSE event is an input to an AND gate, the AND gate is replaced by the HOUSE event.
5. If a true HOUSE event is input to an OR gate that is the fault tree top gate, the HOUSE event satisfies the entire tree and cut sets are not generated.

5.1.3.6 Treatment of Complemented Events. Complemented (or NOT) events are used to indicate the probability of an event not occurring. The probabilities for an event and its complement must always add to 1.0. In general, complemented events are treated the same as other basic events. Since an event and its complement are mutually exclusive (i.e., both cannot occur at the same time), a

check of the cut sets is performed to eliminate cut sets containing an event and its complement.

5.1.3.7 Treatment of NOT Gates. NOT gates are used to indicate the complement of a logical combination of events. NOT gates are processed by using the laws of Boolean algebra to convert the complemented gates to non-complemented gates with complemented inputs. The following rules apply:

1. A NOT OR gate converts to an AND gate with all of the inputs complemented.
2. A NOT AND gate converts to an OR gate with all of the inputs complemented.

5.1.4 Cut Set Generation

The cut set generation algorithm employed in IRRAS 2.0 is a simple top-down algorithm. Each gate is expanded starting with the top gate and proceeding until all non-independent gates are resolved. If available memory is consumed, this expansion process is alternated with a supersetting operation. Once all non-independent gates are processed, the resulting cut sets are passed to a routine to expand the independent gates. The resulting cut sets are written to a disk file.

5.1.5 Supersetting

The fault tree reduction process includes two basic functions. First, the gates are expanded generating all the boolean indicated cut sets (BIC's). Second, the BIC's are reduced by supersetting. This supersetting process involves the elimination of cut sets according to the following rule where A and B are boolean variables:

$$A * B + A = A$$

Any cut set which is a superset of another cut set is eliminated, since the subset is sufficient to satisfy the cut set. This process uses a binary bit algorithm to perform the reduction.

5.2 Minimal Cut Set Upper Bound

The fault tree quantification process is performed in two steps: (1) calculation of individual cut set values and (2) calculation of the minimal cut set upper bound from the cut set values.

The individual cut set probabilities are determined by multiplying the probabilities of the applicable basic events.

$$C_i = BE_1 * BE_2 * BE_3 * \dots * BE_n \quad (6)$$

where

C = probability of cut set i, and

BE = probability of failure of the nth basic event in the
ith cut set.

The minimal cut set upper bound calculation is an approximation to the union of the minimal cut sets for the fault tree. The equation for the minimal cut set upper bound is

$$MC = 1 - \prod_{i=1}^m (1 - C_i) \quad (7)$$

where

MC = minimal cut set upper bound,

C = probability of the ith cut set, and

m = number of minimal cut sets in the fault tree.

The min cut upper bound is always less than or equal to 1. The input values for the min cut upper bound are probabilities. Thus, if an item in a cut set is an initiating event with a frequency greater than 1, the results from the min cut upper bound may be wrong. In IRRAS, if the cut set probabilities are very small, e.g. less than 1.E-16, the min cut upper bound result will not agree with the summation of the cut set probabilities. This due to the calculations with very small numbers.

The min cut upper bound works well with fault trees containing only AND and OR gates without complemented events or NOT gates. With noncoherent fault trees, trees that contain NOT gates and/or complemented events, the min cut upper bound can produce results which can be incorrect. The magnitude of the error will depend upon the structure of the tree. In such cases, other calculational techniques should be used such as the SIGPI algorithm [9]. In most cases, the min cut upper bound will produce reliable results.

5.3 How Are the Sequence Cut Sets Determined?

The sequence minimal cut sets are determined from the cut sets for the fault trees that make up the top events in the sequence. The sequence is defined in terms of failed and successful fault tree top events. The minimal cut sets for each of the fault trees making up the sequence must be generated before the sequence cut sets can be determined.

The first step in generating the sequence cut sets is the comparison of cut sets for the failed and successful fault trees.

This comparison is used to remove impossible cut sets from the failed fault tree cut set lists. Impossible cut sets come about due to the dependence among the fault trees making up the sequence. For example, two systems depend on the same electric power supply. If one of the systems is successful and the other is failed, then it is known that the failed system did not fail due to electric power failures. The cut sets that represent electric power failure should then be removed from the cut set list for the failed system. This is what IRRAS 2.0 does.

The next step is to multiply the cut sets for the failed top events together. This represents the AND logic of the accident sequence (i.e., top event A is failed AND top event B is failed AND top event C ...). This process rapidly fills up the available storage space in memory; therefore, this step is done in stages. Between each stage, a supersetting routine is used to eliminate non-minimal cut sets and reduce the storage requirements. After the cut sets are all multiplied together and superseded, the minimal cut sets are quantified using the same algorithms employed in quantifying the fault tree cut sets.

5.4 Exploring Monte Carlo and Uncertainty

The uncertainty analysis option allows the user to calculate the uncertainty associated with the minimal cut sets currently loaded. To use this option, the user must have previously loaded or generated the cut sets and loaded the component reliability information and distribution data. Sections 3.4 and 3.5 explain these processes.

5.4.1 Basic Uncertainty Information

The Monte Carlo procedure computes the probability distribution of a fault tree top event or accident sequence using the assigned probability distributions for each basic event given in the minimal cut sets. By using the probability distributions for each independent basic event, the data variability and uncertainty in the system unavailability can be better modeled. Note that each basic event is assumed to be independent with no correlation between events except when assigned to a correlation class. Within the correlation class, complete correlation is assumed.

The first step in the process of computing the uncertainty in the minimal cut set upper bound is to provide a measure of the uncertainty for each basic event contained in the minimal cut sets. This requires a mean probability, and an error factor for the probability of failure for the lognormal probability density function indicated for each basic event. The mean value, m , can be expressed as:

$$m = e^{\mu + \sigma^2/2} \quad (8)$$

where μ (mu) is the mean and σ (sigma) is the standard deviation of the underlying normal distribution. Likewise, the 90% error factor (ef) for the lognormal distribution is given by:

$$ef = e^{1.645\sigma} \quad (9)$$

where 1.645 is the 95th percentile of the distribution. Thus, a 90% probability interval for the lognormal distribution is:

$$\left[e^{\mu - 1.645\sigma}, e^{\mu + 1.645\sigma} \right] \quad (10)$$

The uncertainty procedure computes the minimal cut set upper bound for a set of random samples from the uncertainty distributions of the basic events. As shown in Reference 4, this is given by:

$$\bar{A} = 1 - \prod_{j=1}^m \left(1 - \prod_{i=1}^n P_{ij} \right) \quad (11)$$

where

$\bar{A}$ = The minimal cut set upper bound,

P_{ij} = The probability of failure for each basic event in the cut set,

i = The index for each event in a cut set,

n = The number of events in a particular cut set,

j = The index for each cut set in the fault tree or sequence, and.

m = The number of minimal cut sets in the system fault tree or accident sequence.

After calculating the minimal cut set upper bound, the uncertainty procedure computes the first four moments of the distribution and the 5th, 50th, mean, and 95th percentile values.

is approximately three, the normal distribution is a good approximation. A zero skewness value indicates a symmetric distribution; a negative skewness indicates a long left tail, while a positive value indicates a long right tail. If the kurtosis is greater than three, the distribution is more peaked than normal. However, if the value is less than three, the distribution is flatter than normal.

5.4.2 Supported Distributions

At the present time, the following distributions are supported -- Beta, Gamma, Normal, Lognormal, Exponential, and Uniform. These distributions are described by certain statistical parameters. The code requires that the user input the mean of the distribution and one additional parameter, if necessary. Table 4 summarizes this information for each of the supported distributions. The supported distributions are described in the following sections.

»Distribution«	»Code«	»First Parameter«	»Second Parameter«
Lognormal	L	Mean	95% Error Factor
Normal	N	Mean	Standard Deviation
Beta	B	Mean	B in Beta(A,B)
Gamma	G	Mean	R in Gamma(R)
Exponential	E	Mean	None
Uniform	U	Mean	Upper End Point

Table 4. Distributions and Parameters

5.4.2.1 Beta Distribution. The parameters of the Beta distribution are A and B. The probability density function is given by

$$f(x) = \text{Beta}(A,B) x^{A-1} (1-x)^{B-1} \quad (17)$$

where Beta(A,B) is the Beta function. In IRRAS, the value in the uncertainty distribution is B. The parameter A is calculated from the mean value by the formula

$$A = M * B / (1-M) \quad (18)$$

where M is the mean probability of failure. The mean of the Beta distribution is $M = A/(A+B)$. Note that the mean of the Beta distribution is between 0 and 1.

5.4.2.2 Gamma Distribution. The parameters of the Gamma distribution are R and A. The probability density function is given by

The moments are calculated as a basis for comparison of the calculated distribution with other distributions.⁵ From the first four moments, the sample mean, sample variance, coefficient of skewness, and coefficient of kurtosis can be calculated. To establish some standard notation, the following symbols are used:

n = the number of samples calculated.

x_i = i th data value for $i = 1, 2, 3, \dots n$.

The sample mean, given as $\bar{x}$, can be defined as

$$\bar{x} = \frac{\sum_{i=1}^n x_i}{n} \quad (12)$$

and the sample variance, given as

$$s^2 = \frac{\sum_{i=1}^n x_i^2 - n\bar{x}^2}{n} \quad (13)$$

The m th moment about the mean is next defined in general as

$$\mu_m = \frac{\sum (x_i - \bar{x})^m}{n} \quad (14)$$

Thus, from the third moment, the coefficient of skewness is

$$\beta_1 = \frac{\mu_3}{s^3} \quad (15)$$

and from the fourth moment, the coefficient of kurtosis is

$$\beta_2 = \frac{\mu_4}{s^4} \quad (16)$$

where S = the square root of s^2 .

The coefficient of skewness and the coefficient of kurtosis are generally used as a measure for comparison with the normal distribution. If the skewness is close to zero while the kurtosis

$$f(x) = \frac{A^R}{\Gamma(R)} x^{R-1} e^{-Ax} \quad (19)$$

where $\Gamma(R)$ is the Gamma function. In IRRAS the value in the uncertainty distribution is R . A is calculated from the mean value by the formula

$$A = R / M \quad (20)$$

since the mean is $M = R/A$.

5.4.2.3 Exponential Distribution. The mean probability is the parameter for this distribution.

5.4.2.4 Normal Distribution. The value in the uncertainty distribution in IRRAS is the standard deviation of the distribution.

5.4.2.5 Lognormal Distribution. The parameters are the mean of the lognormal distribution and the upper 95% error factor.

5.4.2.6 Uniform Distribution. The mean of this distribution is $M = (A+B)/2$. The value in the uncertainty distribution in IRRAS is B , the right (upper) endpoint of the distribution. A is calculated by the equation $A = 2 * M - B$.

5.5 What Are Importance Measures?

IRRAS 2.0 calculates three different basic event importance measures. These are the Fussell-Vesely importance, the Risk Reduction Ratio, and the Risk Increase Ratio. These importance measures are calculated for each basic event with respect to the fault tree minimal cut set upper bound and the accident sequence frequency.

The Fussell-Vesely importance is an indication of the fraction of the minimal cut set upper bound (or sequence frequency) that involves the basic event of concern. It is calculated by determining the minimal cut set upper bound (sequence frequency) with the basic event failure probability at its true value and again with the basic event failure probability set to zero. The difference between these two results is divided by the true minimal cut set upper bound to obtain the Fussell-Vesely importance. In equation form this is

$$F-V = [F(x) - F(0)] / F(x) \quad (21)$$

where

F-V = Fussell-Vesely importance,

F(x) = minimal cut set upper bound (sequence frequency)
evaluated with the basic event failure probability at
its true value, and

F(0) = minimal cut set upper bound (sequence frequency)
evaluated with the basic event failure probability set
to zero.

The Risk Reduction Ratio is an indication of how much the results would be reduced if the specific event were totally reliable (failure probability equals zero). The Risk Reduction Ratio is determined by evaluating the fault tree minimal cut set upper bound (or the sequence frequency) with the basic event set to its true value and dividing it by the minimal cut set upper bound (sequence frequency) calculated with the basic event set to zero. In equation form this is

$$RR = F(x) / F(0) \quad (22)$$

where

RR = Risk Reduction Ratio,

F(x) = minimal cut set upper bound (sequence frequency)
evaluated with the basic event probability at its true
value, and

F(0) = minimal cut set upper bound (sequence frequency)
evaluated with the basic event probability set to zero.

The Risk Increase Ratio is an indication of how much the results would go up if the specific event was totally unreliable (failure probability equal to 1.0). The Risk Increase Ratio is determined by evaluating the minimal cut set upper bound (sequence frequency) with the basic event failure probability set to 1.0 and dividing it by the minimal cut set upper bound evaluated with the basic event failure probability set to its true value. In equation form this is

$$RI = F(1) / F(x) \quad (23)$$

where

RI = Risk Increase Ratio,

F(1) = minimal cut set upper bound (sequence frequency)
evaluated with the basic event failure probability set
to 1.0, and

$F(x)$ = minimal cut set upper bound (sequence frequency)
evaluated with the basic event failure probability set
at its true value.

The Birnbaum importance measure is calculated in place of the Fussell-Vesely importance measure when intervals are selected instead of ratios. The Birnbaum importance is an indication of the sensitivity of the minimal cut set upper bound (or sequence frequency) with respect to the basic event of concern. It is calculated by determining the minimal cut set upper bound (or sequence frequency) with the basic event of concern set to 1.0 and again with the basic event set to 0.0. The difference between these two values is the Birnbaum importance. In equation form, this is

$$B = F(1) - F(0) \quad (24)$$

where

B = Birnbaum importance,

$F(1)$ = minimal cut set upper bound (sequence frequency)
evaluated with the basic event failure probability set
to 1.0, and

$F(0)$ = minimal cut set upper bound (sequence frequency)
evaluated with the basic event failure probability set
to 0.0.

The Risk Reduction Interval and the Risk Increase Interval indicate the same characteristics as their ratio counterparts but reflect actual minimal cut set upper bound (sequence frequency) levels instead of ratios.

The Risk Reduction Interval is calculated by taking the difference between the nominal value and the function evaluated at 0.

$$RRI = F(x) - F(0) \quad (25)$$

The Risk Increase Interval is calculated by taking the difference between the function evaluated at 1.0 and the nominal value.

$$RII = F(1) - F(x) \quad (26)$$

The ratio definitions are dimensionless and consider only relative changes. The interval definitions account for the actual risk levels which 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.

6. REFERENCES .

1. Borland International, Inc., "Sidekick," first edition March 1985.
2. W. E. Vesely et al., Fault Tree Handbook, NUREG-0492, January 1981.
3. R. B. Worrell and D. W. Stack, A SETS User's Manual for the Fault Tree Analyst, NUREG/CR-0465, SAND77-2051, November 1978.
4. J. B. Fussell, "How to Hand Calculate System Reliability and Safety Characteristics," IEEE Transactions on Reliability, R-24, 3, August 1975.
5. E. J. McGrath and D. C. Irving, "Variance Reduction," Techniques for Efficient Monte Carlo Simulation, III, ORNL-RSIC-38, April 1975.
6. G. W. Snedecor and W. G. Cochran, Statistical Methods. 6th edition, Ames, Iowa: Iowa State University Press, 1967.
7. W. J. Conover, Practical Nonparametric Statistics, New York: John Wiley & Sons, 1971.
8. Corynen, G. C., A Fast Bottom up Algorithm for Computing the Cutsets of Non-Coherent Fault Trees, NUREG/CR-5242, October 1988.
9. Patenaude, C. J., SIGPI: A User's Manual for Fast Computation of Probabilistic Performance of Complex Systems, NUREG/CR-4800, Lawrence Livermore National Laboratory, May 1987.

APPENDIX A

HARDWARE REQUIREMENTS AND INSTALLATION PROCEDURE

A-1/A-2

APPENDIX A

HARDWARE REQUIREMENTS AND INSTALLATION PROCEDURE

The IRRAS 2.0 system requires the following hardware configurations:

- IBM-PC/XT/AT PS2 or 100% compatible 640K main memory
- A hard disk with a minimum of 7.5 mb free
- Math co-processor (mandatory)
- 16 color enhanced monitor
- graphics input devices - keyboard; 2-, 3-, or 4-button mouse; or Summagraphics tablet

If an enhanced graphics adapter is used, it must have the memory expansion option to extend the standard 4 colors to 16. This option is an upgrade to the IBM EGA board, but is usually standard on boards manufactured by other vendors. IRRAS 2.0 does not support the 4-color mode on the EGA adapter.

The recommended configuration contains an enhanced color monitor and adapter which has been upgraded to allow the use of 16 colors and a mouse as the graphics input device. The system will operate with the standard monitor; however, the graphics options will not work. The keyboard can serve as the graphics input device but is not nearly as user-friendly as the mouse. The IRRAS system requires the math co-processor for operation of the graphics and floating point deimal calculations. IRRAS will not run if a math co-processor is not present.

NOTE: The IRRAS 2.0 system requires the above specified amount of random access memory (RAM). This memory must be available for the program and not used by memory resident programs, such as "Side-Kick". Also certain memory resident programs may interfere with the execution of IRRAS 2.0. These should be unloaded before using IRRAS 2.0. The MS/DOS command "CHKDSK" can be used to display the amount of memory available on your machine.

To install the system on your hard drive C, put the #1 diskette in drive A and type A:INSTALL C: <Enter>. For PS2 users with an external 5 1/4" drive, use the MS/DOS Command ASSIGN to assign your external drive as the A drive. You will be prompted when you should insert the remaining disks. Be certain to insert the disks in the order specified in the installation procedure. The directories \IRRAS2, \IRRAS2\DEMO, and \HALO will be created and loaded with the appropriate software.

NOTE: You must have at least 7.5 megabytes of disk storage on your destination disk before installing IRRAS. After installation, IRRAS 2.0 consumes 4.5 megabytes of disk space.

After the installation is completed, you need to locate and modify your CONFIG.SYS file. This file is in the root directory. If you do not have a file by this name, you must create one. In either case, the following parameters must be included in the file, if not already present.

```
FILES=20
BUFFERS=15
DEVICE=ANSI.SYS
```

The device driver ANSI.SYS must also be copied from the DOS directory to the root directory if it is not already there.

The installation procedure will create in the \IRRAS2 directory batch procedures, IRRAS2.BAT, IRRAS2X.BAT, and IRRAS20.BAT for executing IRRAS 2.0. You should move these procedures to your batch directory in the executable path or to the root directory or add \IRRAS2 to your path. These procedures may be used directly or adapted to meet your specific needs. If you choose not to modify the procedure, the following format is used to execute IRRAS 2.0. Type:

```
IRRAS? @ * <Enter>
```

where

?	=	20	to execute IRRAS 2.0 from the \IRRAS2 directory only (for computers with operating systems older than DOS 3.0).
		2	to execute IRRAS 2.0 from any directory (must have DOS 3.0 or newer).
		2X	to execute IRRAS 2.0 from any directory without bringing the graphics package into memory (frees up 80K of memory for analyzing larger problems, must have DOS 3.0 or newer).
@	=	M	to use a Mouse Systems compatible mouse as the graphics input device.
		MS	to use a Microsoft compatible mouse as the graphics input device.
		T	to use a graphics tablet as the graphics input device.
		blank	to use the keyboard as the graphics input device.

- * = blank if the graphics input device is a bus mouse or is connected to communications port 1 or is the keyboard.
- 2 if the graphics input device is connected to communications port 2.

The IRRAS20 execution command loads the graphics package and the analysis package into memory. When IRRAS 2.0 is exited, 80K of memory remains tied up by the IRRAS 2.0 graphics until the computer is rebooted. This command must be used on computers with DOS older than 3.0.

For computers using DOS 3.0 or newer, the IRRAS2 command may be used. This command allows execution from any directory or drive.

If graphics are not going to be used in the current session, then IRRAS2X can be used. This lets the user perform all aspects of analysis that do not require going into the graphics and provides 80K more memory for the calculations. When IRRAS 2.0 is exited, no memory is tied up by IRRAS 2.0. IRRAS2X is only executable on computers using DOS 3.0 or newer.

This completes the installation of the IRRAS 2.0 software. The user must now insure that the proper graphics input device is hooked up and ready for use. When this is done, the IRRAS 2.0 system is ready for use. Refer to Section 4.6 for a discussion on defining constants for your configuration.

APPENDIX B
DATA BASE RELATIONS

APPENDIX B

DATA BASE RELATIONS

This appendix contains a detailed description of each relation used by IRRAS 2.0. It contains the field names and their characteristics. This appendix also defines which fields serve as primary and alternate keys for the relations.

Both permanent and temporary relations are included. Permanent relations have non-zero file numbers and have .DAT, .IDX, and .BLK disk files associated with them. Temporary relations have a file number of zero and do not have any disk files.

RECORD REPORT FOR IRRAS

--- 1 ---

Record Name : EventTree
Description : Event Tree records
Fields : 8
Byte Length : 84
File Number : 4
Data File : EVENTREE.DAT
Index File : EVENTREE.IDX
Block File : EVENTREE.BLK

Field Name Repeat	Description Type	Length Decimals	Key Type K/Nod N/Cr	Sub-Fields
Des 1	Event tree description 1 (Alphanumeric)	60	Non-keyed	
EventTID 1	Event tree ID 3 (Integer)	4	Non-keyed	
FamilyID 1	Event's family ID 3 (Integer)	4	Non-keyed	
Graphics 1	Graphics block 12 (Variable Block Data)	4	Non-keyed	
InitEvt 1	Initiating Event number 3 (Integer)	4	Non-keyed	
Name 1	Event tree name 2 (Upper Case Alphanumeric)	16	Non-keyed	
NameKey 1	Family ID + Event Tree Name 0 (Concatenated)	20	Primary 20 5	FamilyID Name
PrimeKey 1	Family ID + Event tree ID 0 (Concatenated)	8	Primary 20 5	FamilyID EventTID

RECORD REPORT FOR IRRAS

--- 2 ---

Record Name : Event
Description : Temporary event record data
Fields : 26
Byte Length : 165

Field Name Repeat	Description Type	Length Decimals	Key Type K/Nod N/Cr	Sub-Fields
CProb 1	Calculated current prob. value 5 (Floating Point Scientific)	5 3	Non-keyed	
CProbType 1	Current probability calc. type 2 (Upper Case Alphanumeric)	1	Non-keyed	
CUncCorr 1	Current Unc. correlation class 3 (Integer)	4	Non-keyed	
CUncType 1	Current uncertainty calc. type 2 (Upper Case Alphanumeric)	1	Non-keyed	
CUncValu 1	Current uncertainty value 5 (Floating Point Scientific)	5 3	Non-keyed	
CValue1 1	Current probability value 1 5 (Floating Point Scientific)	5 3	Non-keyed	
CValue2 1	Current probability value 2 5 (Floating Point Scientific)	5 3	Non-keyed	
CValue3 1	Current probability value 3 5 (Floating Point Scientific)	5 3	Non-keyed	
CompId 1	Event component identifier 2 (Upper Case Alphanumeric)	7	Non-keyed	
CompSys 1	Event component system 2 (Upper Case Alphanumeric)	3	Non-keyed	
CompType 1	Event component type 2 (Upper Case Alphanumeric)	3	Non-keyed	
Des 1	Event description 1 (Alphanumeric)	60	Non-keyed	
EventNum 1	Event Number 3 (Integer)	4	Non-keyed	

RECORD REPORT FOR IRRAS

Field Name	Description	Length	Key Type	Sub-Fields
Repeat	Type	Decimals	K/Nod N/Cr	
FailMode	Event failure mode	2	Non-keyed	
1	2 (Upper Case Alphanumeric)			
Location	Component location	3	Non-keyed	
1	2 (Upper Case Alphanumeric)			
Name	Primary event name	16	Non-keyed	
1	2 (Upper Case Alphanumeric)			
Name2	Secondary event name	16	Non-keyed	
1	2 (Upper Case Alphanumeric)			
Prob	Calculated probability value	5	Non-keyed	
1	5 (Floating Point Scientific)	3		
ProbType	Probability calculation type	1	Non-keyed	
1	2 (Upper Case Alphanumeric)			
Suscept	Susceptability	2	Non-keyed	
16	11 (Boolean) (Y/blank)			
UncCorr	Uncertainty correlation class	4	Non-keyed	
1	3 (Integer)			
UncType	Uncertainty calculation type	1	Non-keyed	
1	2 (Upper Case Alphanumeric)			
UncValue	Uncertainty value	5	Non-keyed	
1	5 (Floating Point Scientific)	3		
Value1	Probability value 1	5	Non-keyed	
1	5 (Floating Point Scientific)	3		
Value2	Probability value 2	5	Non-keyed	
1	5 (Floating Point Scientific)	3		
Value3	Probability value 3	5	Non-keyed	
1	5 (Floating Point Scientific)	3		

RECORD REPORT FOR IRRAS

Field Validation Report

--- 2 ---

Record Name : Event

Description : Temporary event record data

2) CProbTyp Current probability calc. type (Upper Case Alphanumeric)

Enumerated List - 11 value(s)

1 2 3 4 5 6 7 8 9 F T

19) ProbType Probability calculation type (Upper Case Alphanumeric)

Enumerated List - 11 value(s)

1 2 3 4 5 6 7 8 9 F T

RECORD REPORT FOR IRRAS

--- 3 ---

Record Name : EventCng
 Description : Event changes
 Fields : 7
 Byte Length : 88
 File Number : 5
 Data File : ECHANGE.DAT
 Index File : ECHANGE.IDX
 Block File : ECHANGE.BLK

Field Name Repeat	Description Type	Length Decimals	Key Type K/Nod N/Cr	Sub-Fields
-----	-----	-----	-----	-----
ChangeID 1	Name of event change group 2 (Upper Case Alphanumeric)	16	Non-keyed	
Date 1	Date of event changes 7 (Date) (YYYY/mm/dd)	10	Non-keyed	
Des 1	Event group change descript. 1 (Alphanumeric)	60	Non-keyed	
EvClass 1	Event class changes 12 (Variable Block Data)	4	Non-keyed	
EvProb 1	Event probability changes 12 (Variable Block Data)	4	Non-keyed	
FamilyID 1	Family identifier 3 (Integer)	4	Alternate 20 5	
PrimeKey 1	Primary key field 0 (Concatenated)	20	Primary 20 5	FamilyID ChangeID

RECORD REPORT FOR IRRAS

--- 4 ---

Record Name : Family
Description : A grouping of PRA data
Fields : 19
Byte Length : 135
File Number : 1
Data File : FAMILY.DAT
Index File : FAMILY.IDX
Block File : FAMILY.BLK

Field Name	Description	Length	Key Type	Sub-Fields
Repeat	Type	Decimals	K/Nod N/Cr	
Des	Family description	60	Non-keyed	
1	1 (Alphanumeric)			
EvAttrib	Event attributes block	4	Non-keyed	
1	12 (Variable Block Data)			
EvDes	Event description block	4	Non-keyed	
1	12 (Variable Block Data)			
EvEmpty	Event empty bit map	4	Non-keyed	
1	12 (Variable Block Data)			
EvFTMap	Event fault tree use map	4	Non-keyed	
1	12 (Variable Block Data)			
EvFTMapS	Size of each event's FT map	4	Non-keyed	
1	3 (Integer)			
EvName1	Event names block	4	Non-keyed	
1	12 (Variable Block Data)			
EvName2	Event names block	4	Non-keyed	
1	12 (Variable Block Data)			
EvProb	Event probability data block	4	Non-keyed	
1	12 (Variable Block Data)			
EvSQMapS	Size of each event's seq map	4	Non-keyed	
1	3 (Integer)			
EvSeqMap	Event sequence use map	4	Non-keyed	
1	12 (Variable Block Data)			

RECORD REPORT FOR IRRAS

Field Name Repeat	Description Type	Length Decimals	Key Type K/Nod N/Cr	Sub-Fields
EvTotal 1	Number of events in block 3 (Integer)	4	Non-keyed	
FTChange 1	Fault tree changes bit map 12 (Variable Block Data)	4	Non-keyed	
FTEmpy 1	Fault tree empty bit map 12 (Variable Block Data)	4	Non-keyed	
FamilyID 1	Family identifier 3 (Integer)	4	Primary 20 5	
Name 1	Family name 2 (Upper Case Alphanumeric)	16	Primary 20 5	
SQChange 1	Sequence changes bit map 12 (Variable Block Data)	4	Non-keyed	
SQEmpy 1	Sequence empty bit map 12 (Variable Block Data)	4	Non-keyed	
SeqSum 1	Sum of sequences family 5 (Floating Point Scientific)	5 3	Non-keyed	

RECORD REPORT FOR IRRAS

--- 5 ---

Record Name : FaultTree
Description : Fault tree record
Fields : 21
Byte Length : 162
File Number : 2
Data File : FAULTTREE.DAT
Index File : FAULTTREE.IDX
Block File : FAULTTREE.BLK

Field Name Repeat	Description Type	Length Decimals	Key Type K/Nod N/Cr	Sub-Fields
AltCuts 1	Alternate block of cut sets 12 (Variable Block Data)	4	Non-keyed	
Cutsets 1	Block of cut sets 12 (Variable Block Data)	4	Non-keyed	
Des 1	Fault tree description 1 (Alphanumeric)	60	Non-keyed	
FamilyID 1	Associated family identifier 3 (Integer)	4	Non-keyed	
Flags 8	Modification flags 11 (Boolean) (T/F)	1	Non-keyed	
Graphics 1	Fault tree graphics block 12 (Variable Block Data)	4	Non-keyed	
Level 1	Fault tree level 3 (Integer)	1	Non-keyed	
Logic 1	Fault tree logic block 12 (Variable Block Data)	4	Non-keyed	
MinCut 1	Minimum cut sets upper bound 5 (Floating Point Scientific)	5 3	Non-keyed	
Name 1	Fault tree name 0 (Concatenated)	16	Non-keyed	Name1 Name2
Name1 1	First 8 char. of fault tr name 2 (Upper Case Alphanumeric)	8	Non-keyed	

RECORD REPORT FOR IRRAS

Field Name	Description	Length	Key Type	Sub-Fields
Repeat	Type	Decimals	K/Nod N/Cr	
Name2	Last 8 char. of fault tr name	8	Non-keyed	
1	2 (Upper Case Alphanumeric)			
NameKey	Fam ID + FT Name1	12	Primary	FamilyID
1	0 (Concatenated)		20 5	Name1
NumCut	Number of cut sets in Unc. calc	5	Non-keyed	
1	3 (Integer)			
NumEvent	Number of events in Unc. calc.	5	Non-keyed	
1	3 (Integer)			
PrimeKey	Family ID + Fault Tree ID	8	Primary	FamilyID
1	0 (Concatenated)		20 5	TreeID
Quantile	19X4 reals for 0 lvl trees	4	Non-keyed	
1	12 (Variable Block Data)			
SampSize	Monte Carlo sample size	5	Non-keyed	
1	3 (Integer)			
Seed	Random seed used for Unc. calc	5	Non-keyed	
1	3 (Integer)			
TreeID	Fault tree identifier	4	Non-keyed	
1	3 (Integer)			
Uncert	Uncertainty values	5	Non-keyed	
9	5 (Floating Point Scientific)	3		

RECORD REPORT FOR IRRAS

--- 6 ---

Record Name : Sequence
Description : Accident sequence record
Fields : 19
Byte Length : 159
File Number : 3
Data File : SEQ.DAT
Index File : SEQ.IDX
Block File : SEQ.BLK

Field Name Repeat	Description Type	Length Decimals	Key Type K/Nod N/Cr	Sub-Fields
AltCuts 1	Alternate sequence cut set blk. 12 (Variable Block Data)	4	Non-keyed	
Cutsets 1	Sequence cut set block 12 (Variable Block Data)	4	Non-keyed	
Des 1	Sequence description 1 (Alphanumeric)	60	Non-keyed	
EventTID 1	Event tree ID 3 (Integer)	4	Non-keyed	
FamilyID 1	Accident's family ID 3 (Integer)	4	Non-keyed	
Flags 8	Modification flags 11 (Boolean) (T/F)	1	Non-keyed	
JoinKey 1	Family ID + Ev tree ID 0 (Concatenated)	8	Alternate 20 5	FamilyID EventTID
Logic 1	Sequence logic block 12 (Variable Block Data)	4	Non-keyed	
MinCut 1	Sequence min cut set upper bnd 5 (Floating Point Scientific)	5 3	Non-keyed	
Name 1	Sequence name 2 (Upper Case Alphanumeric)	16	Non-keyed	
NameKey 1	Family ID + Sequence Name 0 (Concatenated)	20	Primary 20 5	FamilyID Name

RECORD REPORT FOR IRRAS

Field Name Repeat	Description Type	Length Decimals	Key Type K/Nod N/Cr	Sub-Fields
NumCut 1	Number of cutsets in Unc. calc 3 (Integer)	5	Non-keyed	
NumEvent 1	Number of events in Unc. calc. 3 (Integer)	5	Non-keyed	
PrimeKey 1	Family ID + SeqID 0 (Concatenated)	8	Primary 20 5	FamilyID SeqID
Quantile 1	19X4 reals for 0 level trees 12 (Variable Block Data)	4	Non-keyed	
SampSize 1	Monte Carlo sample size 3 (Integer)	5	Non-keyed	
Seed 1	Random seed used for Unc. calc 3 (Integer)	5	Non-keyed	
SeqID 1	Accident sequence ID 3 (Integer)	4	Non-keyed	
Uncert 9	Sequence uncertainty values 5 (Floating Point Scientific)	5 3	Non-keyed	

RECORD REPORT FOR IRRAS

--- 7 ---

Record Name : UserInfo
 Description : General system user info.
 Fields : 36
 Byte Length : 165
 File Number : 6
 Data File : USERINFO.DAT
 Index File : USERINFO.IDX

Field Name Repeat	Description Type	Length Decimals	Key Type K/Nod N/Cr	Sub-Fields
Blank 1	Blank under name of gate 11 (Boolean) (Y/N)	1	Non-keyed	
Box 1	Box for small text 11 (Boolean) (Y/N)	1	Non-keyed	
EditPath 1	Text editor name (w/path) 2 (Upper Case Alphanumeric)	36	Non-keyed	
EventNam 1	Event name 2 (Upper Case Alphanumeric)	10	Non-keyed	
Fill 1	Fill on or off 11 (Boolean) (Y/N)	1	Non-keyed	
FillC 16	Fill Color 11 (Boolean) (X/blank)	2	Non-keyed	
GateName 1	Gate name 2 (Upper Case Alphanumeric)	10	Non-keyed	
Grid 1	Grid on or off 11 (Boolean) (Y/N)	1	Non-keyed	
HaloPath 1	Halo drive and directory path 2 (Upper Case Alphanumeric)	36	Non-keyed	
IBMFont 1	Printer has IBM font 11 (Boolean) (Y/N)	1	Non-keyed	
LineC 16	Line color 11 (Boolean) (X/blank)	2	Non-keyed	
LineType 1	Line type 3 (Integer)	1	Non-keyed	

RECORD REPORT FOR IRRAS

Field Name Repeat	Description Type	Length Decimals	Key Type K/Nod N/Cr	Sub-Fields
MissionT 1	Mission time for prob. calc. 5 (Floating Point Scientific)	5 3	Non-keyed	
Monitor 1	Monitor type (0-EGA, 1-CGA) 3 (Integer)	2	Non-keyed	
Mouse 1	Mouse type 3 (Integer)	1	Non-keyed	
Multi 1	Multipick option on/off 11 (Boolean) (Y/N)	1	Non-keyed	
N/MHite 1	N/M gate text height 4 (Real)	6 2	Non-keyed	
NameC 16	Name color 11 (Boolean) (X/blank)	2	Non-keyed	
NameHite 1	Name text height 4 (Real)	6 2	Non-keyed	
PCut 1	Cutoff by probability? 11 (Boolean) (Y/N)	1	Non-keyed	
PlotPort 1	Pen plotter comm. port 3 (Integer)	1	Non-keyed	
PrimeNam 1	Primary Ev names(TRUE) or not? 11 (Boolean) (Y/N)	1	Non-keyed	
ProbCut 1	Probability cutoff value 5 (Floating Point Scientific)	5 3	Non-keyed	
Ratios 1	Ratios instead of intervals 11 (Boolean) (Y/N)	1	Non-keyed	
SCut 1	Cutoff by cut set size? 11 (Boolean) (Y/N)	1	Non-keyed	
SFamily 1	Currently selected family 3 (Integer)	4	Non-keyed	
Seed 1	Random number seed 3 (Integer)	5	Non-keyed	

RECORD REPORT FOR IRRAS

Field Name Repeat	Description Type	Length Decimals	Key Type K/Nod N/Cr	Sub-Fields
ShowName 1	Show the name of symbols 11 (Boolean) (Y/N)	1	Non-keyed	
ShowText 1	Show the text 11 (Boolean) (Y/N)	1	Non-keyed	
SizeCut 1	Cutset size cutoff value 3 (Integer)	4	Non-keyed	
TextAlig 1	Text alignment 2 (Upper Case Alphanumeric)	1	Non-keyed	
TextC 16	Name color 11 (Boolean) (X/blank)	2	Non-keyed	
TextHite 1	Test label Height 4 (Real)	6 2	Non-keyed	
UserName 1	User's name 1 (Alphanumeric)	36	Non-keyed	
VSpace 1	Verticle text spacing 4 (Real)	6 2	Non-keyed	
samples 1	Number of samples for unc. cal 3 (Integer)	6	Non-keyed	

RECORD REPORT FOR IRRAS

Field Validation Report

--- 8 ---

Record Name : UserInfo

Description : General system user info.

12)	LineType	Line type	(Integer)

	Minimum Value	->1	
	Maximum Value	->3	
15)	Mouse	Mouse type	(Integer)

	Minimum Value	->1	
	Maximum Value	->2	
17)	N/MHite	N/M Gate text Height	(Real)

	Minimum Value	-> .00	
	Maximum Value	->120.00	
19)	NameHite	Name text Height	(Real)

	Minimum Value	-> .00	
	Maximum Value	->120.00	
21)	PlotPort	Pen plotter comm. port	(Integer)

	Minimum Value	->0	
	Maximum Value	->3	
27)	Seed	Random number seed	(Integer)

	Minimum Value	-> 0	
	Maximum Value	->65533	
31)	TextAlig	Text alignment	(Upper Case Alphanumeric)

	Enumerated List	- 3 value(s)	

	C	L	R

RECORD REPORT FOR IRRAS

33) TextHite Test label Height (Real)

Minimum Value -> .00
Maximum Value -> 120.00

35) VSpace Verticle text spacing (Real)

Minimum Value -> 1.00
Maximum Value -> 10.00

36) samples Number of samples for unc. cal (Integer)

Minimum Value -> 10
Maximum Value -> 5000

RECORD REPORT FOR IRRAS

--- 9 ---

Record Name : Utility
Description : Utility fields
Fields : 49
Byte Length : 576

Field Name	Repeat	Description Type	Length Decimals	Key Type K/Nod N/Cr	Sub-Fields
A1	1	2 (Upper Case Alphanumeric)	1	Non-keyed	
A53	1	Alpha field of length 53 1 (Alphanumeric)	53	Non-keyed	
A54	1	Alpha field of length 54 1 (Alphanumeric)	54	Non-keyed	
B	1	Boolean field (y/n) 11 (Boolean) (Y/N)	1	Non-keyed	
B1	1	Boolean (Y/N) 11 (Boolean) (Y/N)	1	Non-keyed	
B2	1	Boolean (Y/N) 11 (Boolean) (Y/N)	1	Non-keyed	
Comp	4	+/- fields for seq. rpt 1 (Alphanumeric)	1	Non-keyed	
CutId	1	Cut sets identification 3 (Integer)	4	Non-keyed	
CutPcnt	1	Cumulat. percent of cutsets 4 (Real)	5 1	Non-keyed	
EName1	1	Temporary event name 2 (Upper Case Alphanumeric)	17	Non-keyed	
EName2	1	Temporary event name 2 (Upper Case Alphanumeric)	17	Non-keyed	
ETime	1	Elapsed time 13 (Time)(with microseconds)	12	Non-keyed	
FTName1	1	Name for FT editor 1 2 (Upper Case Alphanumeric)	17	Non-keyed	

RECORD REPORT FOR IRRAS

Field Name Repeat	Description Type	Length Decimals	Key Type K/Nod N/Cr	Sub-Fields
FTName2 1	Name for FT editor 2 2 (Upper Case Alphanumeric)	17	Non-keyed	
FTName3 1	2 (Upper Case Alphanumeric)	4	Non-keyed	
FTTitle 1	Fault Tree Name for Rpt Title 1 (Alphanumeric)	30	Non-keyed	
FV 1	Fussel Vesley value for Rpt 5 (Floating Point Scientific)	5 3	Non-keyed	
FailRate 1	Failure Rate prob. for Rpt 5 (Floating Point Scientific)	5 3	Non-keyed	
FamTitle 1	Family Name for Report Title 1 (Alphanumeric)	25	Non-keyed	
FileID 1	File ID for 1.0 files 2 (Upper Case Alphanumeric)	12	Non-keyed	
FileID8 1	8 char file name - no ext. 2 (Upper Case Alphanumeric)	8	Non-keyed	
FileName 1	file name 1 (Alphanumeric)	32	Non-keyed	
Freq 1	Frequency of cutset 5 (Floating Point Scientific)	3 1	Non-keyed	
GateName 1	Gate name for Fault Tree Rpt 1 (Alphanumeric)	16	Non-keyed	
GateType 1	Gate type for Fault Tree Rpt 1 (Alphanumeric)	4	Non-keyed	
I1 1	3 (Integer)	1	Non-keyed	
I2 1	Integer of length 2 3 (Integer)	2	Non-keyed	
LC1 1	1 (Alphanumeric)	1	Non-keyed	

RECORD REPORT FOR IRRAS

Field Name	Description	Length	Key Type	Sub-Fields
Repeat	Type	Decimals	K/Nod N/Cr	
Mean	Uncertainty mean value	5	Non-keyed	
1	5 (Floating Point Scientific)	3		
MinCut	Minimum cut sets upper bound	5	Non-keyed	
1	5 (Floating Point Scientific)	3		
MinCutSm	Minimum cut set upper bound sum	5	Non-keyed	
1	5 (Floating Point Scientific)	3		
NumCut	Num of cut sets total	4	Non-keyed	
1	3 (Integer)			
QEvents	Qualified Events	4	Non-keyed	
1	3 (Integer)			
RiskAch	Risk Achievmnt value for Rpt	5	Non-keyed	
1	5 (Floating Point Scientific)	3		
RiskRed	Risk Reduction value for Rpt	5	Non-keyed	
1	5 (Floating Point Scientific)	3		
SizeBins	Cut set size bins	5	Non-keyed	
12	3 (Integer)			
SliceCut	Slice cut set frequency	5	Non-keyed	
1	5 (Floating Point Scientific)	3		
SliceNum	Num of cut sets in this slice	4	Non-keyed	
1	3 (Integer)			
SliceP	This slice percent	7	Non-keyed	
1	4 (Real)	2		
T1		1	Non-keyed	
1	2 (Upper Case Alphanumeric)			
TempName	Name fields for seq. rpt	16	Non-keyed	
4	2 (Upper Case Alphanumeric)			
Title	Title of the requested report	60	Non-keyed	
1	1 (Alphanumeric)			

RECORD REPORT FOR IRRAS

Field Name Repeat	Description Type	Length Decimals	Key Type K/Nod N/Cr	Sub-Fields
-----	-----	-----	-----	-----
TotGates 1	Total gates for the fault tr. 3 (Integer)	6	Non-keyed	
TotPcnt 1	Total percent of cut set 4 (Real)	5 1	Non-keyed	
TreeName 1	Name of the fault tree 2 (Upper Case Alphanumeric)	16	Non-keyed	
Uncert 9	Uncertainty values 5 (Floating Point Scientific)	5 3	Non-keyed	

RECORD REPORT FOR IRRAS

Field Validation Report

--- 9 ---

Record Name : Utility
Description : utility fields

29) MainOptn Main menu options (Upper Case Alphanumeric)

Enumerated List - 8 value(s)

BUILD CREATE EXIT FAULT MODIFY REPORT
SEQUENCE UTILITY

APPENDIX C
IRRAS 2.0 REPORTS

APPENDIX C

IRRAS 2.0 REPORTS

This appendix displays all the reports contained in the REPORT on Data Base module of IRRAS 2.0. Each report can be displayed on the screen, printed out on a printer, or sent to a file on the hard disk. Some of the reports, especially the basic event reports, may be fairly large. These reports can take some time to compile and print. The printouts may be interrupted periodically for further data compilation prior to completion.

REPORT 1, BASIC EVENTS SUMMARY REPORT

BASIC EVENTS SUMMARY REPORT

Family: DEMO FAMILY

<u>Event Number</u>	<u>Primary Name</u>	<u>Secondary Name</u>	<u>Component Type</u>	<u>ID</u>	<u>System</u>	<u>Loca- tion</u>	<u>Fail Mode</u>
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)

- (1) Event Number. The position of the basic event in the listing.
- (2) Primary Name. The basic event name used for most work, typically the PRA name.
- (3) Secondary Name. An alternate basic event name to be used for special purposes, such as interacting with plant operators.
- (4) Component Type. The component type classification for the basic event, such as MDP for motor-driven pump.
- (5) Component ID. The specific designator for the component from the plant drawings, nameplate, etc.
- (6) Component System. The system classification for the basic event, such as CCS for Containment Cooling System.
- (7) Location. The location designator for the basic event. This may be a fire zone, room number, grid coordinates, etc.
- (8) Failure Mode. The failure mode designator for the basic event, such as FTO for failure to open.

REPORT 2, BASIC EVENTS PROBABILITY REPORT

BASIC EVENTS PROBABILITY REPORT

Family: DEMO FAMILY

Event Number	Primary Name	Calculation Type	Mean Probability	Lamda	Tau
(1)	(2)	(3)	(4)	(5)	(6)

- (1) Event Number. The position of the basic event in the listing.
- (2) Primary Name. The basic event name used for most work, typically the PRA name.
- (3) Calculation Type. The designator for the type of calculation to be used to determine the basic event probability of failure. See Section 3.4.1 for details.
- (4) Mean Probability. The results of the calculation of the mean failure probability using the calculation type and failure data.
- (5) Lamda. The failure rate (per hour) to be used in the calculation of the basic event failure probability.
- (6) Tau. The time (in hours) to be used in the calculation of the basic event failure probability.

REPORT 3, BASIC EVENTS UNCERTAINTY VALUES REPORT

BASIC EVENTS UNCERTAINTY VALUES REPORT

Family: DEMO FAMILY

Event Number	Primary Name	Distribution Type	Mean Probability	Uncertainty Value	Correlation Class
(1)	(2)	(3)	(4)	(5)	(6)

- (1) Event Number. The position of the basic event in the listing.
- (2) Primary Name. The basic event name used for most work, typically the PRA name.
- (3) Distribution Type. The type of distribution describing the uncertainty of failure probability for the basic event. Currently, only lognormal distributions are allowed.
- (4) Mean Probability. The results of the calculation of the mean failure probability using the calculation type and failure data.
- (5) Uncertainty Value. The parameter, in addition to the mean probability, used to define the distribution for the uncertainty. For the lognormal distribution, this is the error factor.
- (6) Correlation Class. The designator for the correlation class assigned to the basic event. See Section 3.4.2 for more details.

REPORT 4, FAULT TREE LOGIC REPORT

FAULT TREE LOGIC REPORT

Family: DEMO FAMILY
Fault Tree: CCS

Gate Name	Type	Inputs
(1)	(2)	(3)

- (1) Gate Name. The assigned name (up to 16 characters) of the fault tree gate.
- (2) Type. The gate type, such as AND, OR, TRAN.
- (3) Inputs. The names of the gates and basic events attached to the bottom of this gate.
- (4) Transfer gates, n/m gates, and complimented gates are not expanded.

REPORT 5, FAULT TREE EXPANDED LOGIC REPORT

FAULT TREE EXPANDED LOGIC REPORT

Family: DEMO FAMILY

Fault Tree: CCS

Gate Name	Type	Inputs
(1)	(2)	(3)

(1) Gate Name. The assigned name (up to 16 characters) of the fault tree gate.

(2) Type. The gate type, such as AND, OR, TRAN.

(3) Inputs. The names of the gates and basic events attached to the bottom of this gate.

(4) Transfer gates are expanded if they exist.

REPORT 6, FAULT TREE MODIFIED LOGIC REPORT

FAULT TREE MODIFIED LOGIC REPORT

Family: DEMO FAMILY
Fault Tree: CCS

Gate Name	Type	Inputs
(1)	(2)	(3)

- (1) Gate Name. The assigned name (up to 16 characters) of the fault tree gate.
- (2) Type. The gate type, such as AND, OR, TRAN.
- (3) Inputs. The names of the gates and basic events attached to the bottom of this gate.

REPORT 7, FAULT TREE IMPORTANCE MEASURES REPORT

FAULT TREE IMPORTANCE MEASURES REPORT

Family: DEMO FAMILY
Fault Tree: CCS
(Sorted by Fussell-Vesely)

<u>Event Name</u>	<u>Probability of Failure</u>	<u>Fussell- Vesely Importance</u>	<u>Risk Reduction Ratio</u>	<u>Risk Increase Ratio</u>
(1)	(2)	(3)	(4)	(5)

- (1) Event Name. Name of one of the basic events in the fault tree listed in the header.
- (2) Probability of Failure. The mean failure probability of the basic event as calculated by IRRAS 2.0.
- (3) Fussell-Vesely Importance. The fraction of the top event frequency attributable to the basic event listed.
- (4) Risk Reduction Ratio. The ratio of the top event frequencies evaluated with the basic event set to its mean failure probability and zero (0).
- (5) Risk Increase Ratio. The ratio of the top event frequencies evaluated with the basic event set to one (1) and its mean failure probability.

REPORT 8, FAULT TREE CUT SETS REPORT

FAULT TREE CUT SETS REPORT

Family: DEMO FAMILY
Fault Tree: CCS

<u>Cut No.</u>	<u>Size</u>	<u>Cut Sets</u>
(1)	(2)	(3)

- (1) Cut No. The cut set, listed in order from 1 to however many cut sets are in the list.
- (2) Size. The number of basic events in the cut set.
- (3) Cut Sets. The names of the basic events making up the cut set.

REPORT 9, FAULT TREE CUT SETS (QUANTIFICATION) REPORT

FAULT TREE CUT SETS (QUANTIFICATION) REPORT

Family: DEMO FAMILY
Fault Tree: CCS
Mincut Upper Bound 2.120E-002

Cut No.	% Total	% Cut Set	Freq.	Cut Sets
(1)	(2)	(3)	(4)	(5)

- (1) Cut No. The ranking of the cut set according to frequency.
- (2) % Total. The cumulative percentage of the cut sets to the minimal cut set upper bound.
- (3) % Cut Set. The percentage contribution of the cut set to the minimal cut set upper bound.
- (4) Freq. The mean frequency of the cut set.
- (5) Cut Sets. A listing of the basic events making up the cut set.

REPORT 10, FAULT TREE SUMMARY REPORT

FAULT TREE SUMMARY REPORT

Family: DEMO FAMILY

<u>Tree No.</u>	<u>Fault Tree Name</u>	<u>Level</u>	<u>MinCut Upper Bound</u>
(1)	(2)	(3)	(4)

- (1) Tree No. The order that the fault tree appears in the data base.
- (2) Fault Tree Name. The name given to the fault tree when it was entered into the data base.
- (3) Level. The level assigned to the fault tree. See Section 3.3.6 for more on fault tree level assignments.
- (4) MinCut Upper Bound. The point estimate calculation for the fault tree, applicable to zero level fault trees only.

REPORT 11, FAULT TREE UNCERTAINTY VALUES REPORT

FAULT TREE UNCERTAINTY VALUES REPORT

Family: DEMO FAMILY

Tree No.	Fault Tree Name	Random Seed = 3571	Sample Size = 1000
		Uncertainty Values	

(1) (2) (3)

- (1) Tree No. The order that the zero-level fault tree appears in the data base.
- (2) Fault Tree Name. The name given to the fault tree when it was entered into the data base.
- (3) Uncertainty Values. The 5th, median, mean, and 95th percentile values of the top event failure probability distribution from the Monte Carlo uncertainty analysis. Also gives the minimum and maximum sample values, the standard deviation of the distribution, and the minimal cut set upper bound point estimate calculation.

REPORT 12, SEQUENCE LOGIC REPORT

SEQUENCE LOGIC REPORT

Family: DEMO FAMILY

Seq. No.	Sequence Name	Logic
(1)	(2)	(3)

- (1) Seq. No. The order that the sequences appear in the data base.
- (2) Sequence Name. The name that the sequence was given when it was created in the data base.
- (3) Logic. The initiating event and combination of successful and failed fault tree top events that make up the sequence. See Section 3.6 for more on sequence logic.

REPORT 13, SEQUENCE SUMMARY REPORT

SEQUENCE SUMMARY REPORT

Family: DEMO FAMILY

<u>Seq No.</u>	<u>Sequence Name</u>	<u>MinCut Upper Bound</u>
(1)	(2)	(3)

- (1) Seq. No. The order that the sequences appear in the data base.
- (2) Sequence Name. The name that the sequence was given when it was created in the data base. Underneath the name is the description written for the sequence.
- (3) MinCut Upper Bound. The point estimate minimal cut set upper bound calculation result for the sequence. After the last sequence, the total of all the sequences is provided.

REPORT 15, SEQUENCE IMPORTANCE MEASURES REPORT

SEQUENCE IMPORTANCE MEASURES REPORT

Family: DEMO FAMILY
Sequence: 1-LOSP-ECS
(Sorted by Fussell-Vesely)

<u>Event Name</u>	<u>Probability of Failure</u>	<u>Fussell- Vesely Importance</u>	<u>Risk Reduction Ratio</u>	<u>Risk Increase Ratio</u>
(1)	(2)	(3)	(4)	(5)

- (1) Event Name. Name of one of the basic events in the sequence listed in the header.
- (2) Probability of Failure. The mean failure probability of the basic event as calculated by IRRAS 2.0.
- (3) Fussell-Vesely Importance. The fraction of the sequence frequency attributable to the basic event listed.
- (4) Risk Reduction Ratio. The ratio of the sequence frequencies evaluated with the basic event set to its mean failure probability and zero (0).
- (5) Risk Increase Ratio. The ratio of the sequence frequencies evaluated with the basic event set to one (1) and its mean failure probability.

REPORT 16, SEQUENCE CUT SETS REPORT

SEQUENCE CUT SETS REPORT

Family: DEMO FAMILY
Sequence: 2-LOSP-ECS-CCS

<u>Cut No.</u>	<u>Size</u>	<u>Cut Sets</u>
(1)	(2)	(3)

- (1) Cut No. The order in which the cut sets are listed.
- (2) Size. The number of basic events making up the cut set.
- (3) Cut Sets. A listing of the basic events (and the initiating event) making up the cut set.

REPORT 17, SEQUENCE CUT SETS (QUANTIFICATION) REPORT

SEQUENCE CUT SETS (QUANTIFICATION) REPORT

Family: DEMO FAMILY
Sequence: 2-LOSP-ECS-CCS
Mincut Upper Bound 4.834E-002

Cut No.	% Total	% Cut Set	Freq.	Cut Sets
(1)	(2)	(3)	(4)	(5)

- (1) Cut No.. The ranking of the cut set according to frequency.
- (2) % Total. The cumulative percentage of the cut sets to the minimal cut set upper bound.
- (3) % Cut Set. The percentage contribution of the cut set to the minimal cut set upper bound.
- (4) Freq.. The mean frequency of the cut set.
- (5) Cut Sets. A listing of the basic events making up the cut set.

REPORT 18, FAMILY SUMMARY REPORT

FAMILY SUMMARY REPORT

Family Name

Description

(1)

(2)

(1) Family Name. The name assigned to the family when it was created in the data base.

(2) Description. The description entered into the data base for the family. The description may be up to 60 characters long.

1988/01/19

15:04:00

page 1

APPENDIX D
IRRAS 2.0 MENUS AND FORMS

APPENDIX D

IRRAS 2.0 MENUS AND FORMS

This appendix provides discussion on the hierarchal structure of the menus, forms, and options in IRRAS 2.0. Figure D-1 shows the relationships in the hierarchy. The boxes represent major options that bring up menus with more options. Under each menu is listed the options on that menu. The numbers beside the menus correspond to the numbers on the menus that follow.

Figure D-1. Hierarchy of IRRAS 2.0 menus, forms, and options.

Figure D-1. (Continued).

Figure D-1. (Continued).

Figure D-1. (Continued).

Figure D-1. (Continued).

Figure D-1. (Continued).

Figure D-1. (Continued).

CONTINUATION FROM UTILITIES OPTIONS MENU 27

Continued From
Previous Page

- o SETS TO IRRAS
- o EXIT
- o FAULT TREES
- o CUT SETS (FT)
- o SEQ. CUT SETS
- o 2-VERSION 2 TO 1
- o EXIT
- o FAILURE RATES
- o EXIT
- o STORE EVENT PROBABILITY DATA
- o FAULT TREES
- o EXIT
- o LOGIC
- o MODIFIED LOGIC
- o CUT SETS
- o EXIT
- o STORE CUT SETS
- o SEQ. CUT SETST.
- o EXIT
- o STOR CUT SETS

Figure D-1. (Continued).

MENU 1, IRRAS 2.0 Main Menu.

I R R A S - P C	
Integrated Reliability and Risk Analysis System	
Version 2.0	
EXIT	IRRAS 2.0
BUILD	Fault Trees
FAULT	Tree Analysis
CREATE	Event Trees
SEQUENCE	Analysis
MODIFY	Data Base
REPORT	on Data Base
UTILITY	Options
Option... EXIT ...Selected	

ENTERED FROM\EXITS TO: DOS

OPTIONS:

EXIT	Irras 2.0.	Returns the user back to the current directory.
BUILD	Fault Trees.	Calls up Menu 2, IRRAS-PC Fault Tree Graphics System.
FAULT	Tree Analysis.	Calls up Menu 7, IRRAS-PC Fault Tree Analysis System.
CREATE	Event Trees.	Not yet implemented.
SEQUENCE	Analysis.	Calls up Menu 13, IRRAS-PC Sequence Analysis System.
MODIFY	Data Base.	Calls up Menu 16, Edit Data Base Records.
REPORT	on Data Base.	Calls up Menu 23, IRRAS Reports.
UTILITY	Options.	Calls up Menu 27, IRRAS-PC Utilities.

MENU 2, IRRAS-PC Fault Tree Graphics System.

I R R A S - P C
Fault Tree Graphics System

Currently Selected Family —> DEMO FAMILY

E - Exit
S - Select Family
X - eXtract Fault Trees
B - Build Graphics Trees
P - Plot Tree (plotter)
R - Rasterize Tree
D - Define plotter pens
A - Alpha to Graphics

Option... E

ENTERED FROM/EXITS TO:

MENU 1, IRRAS 2.0 Main Menu-Build Fault Trees.

OPTIONS:

E - Exit.	Returns the user to MENU 1, IRRAS 2.0 Main Menu-Build Fault Trees.
S - Select Family.	Calls up MENU 3, Select Family of Fault Trees.
X - eXtract Fault Trees.	Calls up MENU 4, Extract Fault Trees.
B - Build Graphics Trees.	Enters the graphics module for fault tree drawing.
P - Plot Tree (plotter).	Sends a drawing to an HP 7475 pen plotter.
R - Rasterize Tree.	Calls up MENU 5, Rasterize Fault Trees.
D - Define plotter pens.	Allows the user to map the 16 colors in IRRAS 2.0 to the 6 pens on the plotter.
A - Alpha to Graphics.	Calls up MENU 6, Convert Fault Trees.

MENU 3, Select Family of Fault Trees.

```

Select Family of Fault Trees

Currently Selected Family ----> DEMO FAMILY

Option |S|  Exit / Select Family

DEMO FAMILY      Demonstration sample family

NOTE: To select a Family, enter option <S>, move the cursor to the desired
      family, press <Enter>. Use <PgUp> or <PgDn> to display more Families.

```

ENTERED FROM/EXITS TO:

```
MENU 1, IRRAS 2.0 Main Menu-Build Fault Tree\ MENU 2,  
Fault Trees Graphics System-Select Family.  
MENU 1, IRRAS 2.0 Main Menu-Fault Tree Analysis\ MENU 7,  
Fault Tree Analysis System-Select Family.  
MENU 1, IRRAS 2.0 Main Menu-Sequence Analysis\ MENU 13,  
Sequence Analysis System-Select Family.  
MENU 1, IRRAS 2.0 Main Menu-Utility Option\ MENU 27,  
Utility Options-IRRAS/SETS Interface\ MENU 28,  
IRRAS/SETS Interface-Select Family.  
MENU 1, Main Menu-Utility Option \ MENU 27, Utility  
Option-Version 1.0\2.0 Interface, MENU 29,  
Version 1.0\2.0 Interface-Select Family.
```

OPTIONS:

E - Exit.	Returns the user to the calling (previously displayed) menu.
S - Select Family.	Allows the user to change the current family selected.

MENU 4, Extract Fault Trees.

Extract Fault Trees

Currently Selected Family —> DEMO FAMILY

Option|E| Exit / eXtract Tree / Clear extracted Trees / Display extracted Trees

FAULT TREES

CCS	Containment Cooling System fault tree
ECS	Emergency Cooling System fault tree
LOSP	Loss of Offsite Power Initiating Event fault tree

Note : Use <PgUp> and <PgDn> to display more fault trees.

ENTERED FROM/EXITS TO:

MENU 1, Main Menu-Build Fault Trees\ MENU 2, Fault Tree Graphics System-Extract Fault Trees.

OPTIONS:

E - Exit.	Return to MENU 2, IRRAS-PC Fault Tree Graphics System.
X - eXtract Tree.	Calls up list of fault trees in the family that can be picked for extraction and, therefore, used in the graphics.
C - Clear extracted Trees.	Clears the list of trees extracted for use in the graphics module. All .DLS files are deleted from the current directory. Graphics are then only in the data base.
D - Display extracted Trees.	Displays the fault trees that are currently extracted.

MENU 5, Rasterize Fault Trees.

R a s t e r i z e F a u l t T r e e s

Option |R| Exit / Rasterize
 Low / High

← select a Fault Tree below
← LOW resolution selected

CCSECSLOSPFAULT TREES

Note : Use <PgUp> and <PgDn> to display more Fault Trees.
If you do not select a Fault Tree, all will be done.

Note : Use <PgUp> and <PgDn> to display more Fault Trees.
If you do not select a Fault Tree, all will be done.

ENTERED FROM\EXITS TO:

MENU 1, IRRAS 2.0 Main Menu-Build Fault Trees\ MENU 2,
Fault Tree Graphics System-Rasterize Tree.

OPTIONS:

E - Exit.	Return to MENU 2, IRRAS-PC Fault Tree Graphics System.
R - Rasterize.	Calls up the list of fault trees in the family that can be chosen for rasterization and printing on a laser printer.

MENU 6, Convert Fault Trees (Alpha to Graphics Converter).

C o n v e r t F a u l t T r e e s

Option [C] Exit / Convert Fault Tree ← select a Fault Tree below

CCSECS

Note : Use <PgUp> and <PgDn> to display more Fault Trees.

ENTERED FROM/EXITS TO:

MENU 1, IRRAS 2.0 Main Menu-Build Fault Tree\ MENU 2,
Fault Tree Graphics System-Alpha to Graphics

OPTIONS:

E - Exit.	Return to MENU 2, IRRAS-PC Fault Tree Graphics System.
C - Convert Fault Tree.	Creates a .DLS file in the current directory for the highlighted .TRE file.

MENU 7, IRRAS-PC Fault Tree Analysis System.

IRRAS - PC
Fault Tree Analysis System

Currently Selected Family —> DEMO FAMILY

E - Exit

S - Select Family

M - Modify Event Data

A - Analyze Fault Trees

C - Cut Set Editor

F - Fault Tree Editor

D - Display Fault Trees

Option... E

ENTERED FROM/EXITS TO:

MENU 1, IRRAS 2.0 Main Menu-Fault Tree Anlaysis

OPTIONS:

E - Exit.	Returns the user to MENU 1, IRRAS 2.0 Main Menu.
S - Select Family.	Calls up MENU 3, Select Family of Fault Trees.
M - Modify Event Data.	Calls up MENU 8, Set Event Data, for inputting current case basic event data.
A - Analyze Fault Trees.	Calls up MENU 9, Analyze Fault Trees, for fault tree cut set generation, quantification, and uncertainty analysis.
C - Cut Set Editor.	Calls up MENU 10, Cut Set Editor, for editing of cut sets.
F - Fault Tree Editor.	Calls up MENU 11, Fault Tree Editor, for fault tree editing
D - Display Fault Trees.	Calls up MENU 12, Display Fault Tree Data, to show cut sets, importance measures, and uncertainty results.

MENU 8, Set Event Data.

Set Event Data	
Option	<div style="display: flex; justify-content: space-between;"> [E] Exit / Probability / Fault Tree list / Seq. list ← choose Event Class / Reset to Base Case </div>
Name	Description
C-CV-A	CCS Train A pump discharge check valve
C-CV-B	CCS Train B pump discharge check valve
C-MOV-1	CCS suction isolation valve
C-MOV-A	CCS Train A pump discharge isolation valve
C-MOV-B	CCS Train B pump discharge isolation valve
C-PUMP-A	CCS Train A motor-driven pump
C-PUMP-B	CCS Train B motor-driven pump
DG-A	Emergency diesel generator A
DG-B	Emergency diesel generator B
E-CV-A	ECS Train A pump discharge check valve
E-CV-B	ECS Train B pump discharge check valve
E-MOV-1	ECS suction isolation valve

Note: Use <PgUp> or <PgDn> to display more Events.

ENTERED FROM EXITS TO:

```
MENU 1, IRRAS 2.0 Main Menu-Fault Tree Analysis\ MENU 7,  
Fault Tree Anlaysia System-Modify Event Data  
MENU 1, IRRAS 2.0 Main Menu-Sequence Analysis\ MENU 13,  
Sequence Anlaysia System-Modify Event Data
```

OPTIONS:

E - Exit.	Returns the user to MENU 7, IRRAS-PC Fault Tree Analysis System or MENU 13, IRRAS-PC Sequence Analysis System, depending on where it was entered from.
P - Probability.	Allows the user to input or change the current case reliability data for the highlighted event.
F - Fault Tree List.	Displays the fault trees in the family containing the highlighted basic event.
S - Sequence List.	Displays the sequences in the family containing the highlighted basic event.

MENU 8, Set Event Data Continued

- C - Class. Allows the user to make changes to the current case reliability data for a class or group of like basic events.
- R - Release All Changes. Resets the current case reliability data for all basic events in the family to the base case values.

MENU 9, Analyze Fault Trees.

A n a l y z e F a u l t T r e e s													
Currently Selected Family —> DEMO FAMILY													
Option E	Exit / Cut Set Generation / Quantification / Uncertainty calc.												
<table border="1"><thead><tr><th></th><th>FAULT</th><th>TREES</th></tr></thead><tbody><tr><td>c --> CCS</td><td colspan="2">Containment Cooling System fault tree</td></tr><tr><td>c --> ECS</td><td colspan="2">Emergency Cooling System fault tree</td></tr><tr><td>c --> LOSP</td><td colspan="2">Loss of Offsite Power Initiating Event fault tree</td></tr></tbody></table>			FAULT	TREES	c --> CCS	Containment Cooling System fault tree		c --> ECS	Emergency Cooling System fault tree		c --> LOSP	Loss of Offsite Power Initiating Event fault tree	
	FAULT	TREES											
c --> CCS	Containment Cooling System fault tree												
c --> ECS	Emergency Cooling System fault tree												
c --> LOSP	Loss of Offsite Power Initiating Event fault tree												
<p>Note : Changed indicators precede each fault tree with meanings as follows- f-Fault Trees, c-Cut Sets, q-Quantification, u-Uncertainty More change information may be obtained for each tree using <Esc>. Use <PgUp> and <PgDn> to display more fault trees.</p>													

ENTERED FROM/EXIT TO:

MENU 1, IRRAS 2.0 Main Menu-Fault Tree Analysis/ MENU 7,
Fault Tree Analysis System-Analyze Fault Trees

OPTIONS:

E - Exit.	Returns the user to MENU 7, IRRAS-PC Fault Tree Analysis System.
C - Cut Set Generation.	Allows the user to select the cut set generation criteria and generate the minimal cut sets for the fault tree(s) selected. The minimal cut set upper bound will be calculated also.
Q - Quantification.	Allows the user to quantify existing cut sets without regenerating them.
U - Uncertainty Calc.	Allows the user to set the uncertainty analysis parameters and conduct Monte Carlo analysis on the selected fault tree cut sets.

MENU 10, Cut Set Editor.

C u t S e t E d i t o r

Currently Selected Family —> DEMO FAMILY

Option |S| Exit / Select a Fault Tree Cut Set

CCS	Containment Cooling System fault tree
ECS	Emergency Cooling System fault tree
LOSP	Loss of Offsite Power Initiating Event fault tree

Note: To select a Cut Set, enter option <S>, move the cursor to the desired Fault Tree, press <Enter>. Use <PgUp> or <PgDn> to display more Trees.

ENTERED FROM EXIT TO:

MENU 1, IRRAS 2.0 Main Menu-Fault Tree Analysis\ MENU 7,
Fault Tree Analysis System-Cut Set Editor.

OPTIONS:

E - Exit.	Returns the user to MENU 7, IRRAS-PC Fault Tree Analysis System.
S - Select a Fault Tree Cut Set	Allows the user to edit the cut sets.

MENU 11, Fault Tree Editor

F A U L T T R E E
E D I T O R

Currently Selected Family => DEMO FAMILY

Option |F| Exit / Fault Tree Editor <=== Select a Fault Tree Below

F A U L T	T R E E S
CCS	Containment Cooling System fault tree
ECS	Emergency Cooling System fault tree
LOSP	Loss of Offsite Power Initiating Event fault tree

Note : Use <PgUp> and <PgDn> to display more fault trees.

ENTERED FROM EXITS TO:

MENU 1, IRRAS 2.0 Main Menu-Fault Tree Analysis\ MENU 7,
Fault Tree Analysis System-Cut Set Editor

OPTIONS:

E - Exit	Returns the user to MENU 7, IRRAS-PC Fault Tree Analysis System.
F - Fault Tree Editor	Allows the user to edit Fault Trees

MENU 12, Display Fault Tree Data.

Display Fault Tree Data	
Currently Selected Family —> DEMO FAMILY	
Option E Exit / Cut Sets / Importance measures / Uncertainty data	
-----FAULT TREES-----	
CCS	Containment Cooling System fault tree
ECS	Emergency Cooling System fault tree
LOSP	Loss of Offsite Power Initiating Event fault tree
Note : Use <PgUp> and <PgDn> to display more fault trees.	

ENTERED FROM EXITS TO:

MENU 1, IRRAS 2.0 Main Menu-Fault Tree Analysis\ MENU 7,
Fault Tree Analysis System-Display Fault Trees.

OPTIONS:

E - Exit.	Returns the user to MENU 7, IRRAS-PC Fault Tree Analysis System.
C - Cut sets.	Allows the user to review the quantified minimal cut sets for the selected fault tree. Cut sets are sent to the screen only.
I - Importance measures.	Generates and displays the basic event importance measures for the selected fault tree.
U - Uncertainty data.	Displays the uncertainty results for the selected fault tree.

MENU 13, IRRAS-PC Sequence Analysis System.

IRRAS - PC
Sequence Analysis System

Currently Selected Family —> DEMO FAMILY

E - Exit
S - Select Family
M - Modify Event Data
A - Analyze Sequences
C - Cut Set Editor
D - Display Sequences

Option... E

ENTERED FROM EXITS TO:

MENU 1, IRRAS 2.0 Main Menu-Sequence Analysis

OPTIONS:

E - Exit.	Returns the user to MENU 1, IRRAS 2.0 Main Menu.
S - Select Family.	Calls up MENU 3, Select Family of Fault Trees.
M - Modify Event Data.	Calls up MENU 8, Set Event Data, for inputting current case basic event data.
A - Analyze Sequences.	Calls up MENU 14, Analyze Sequences, for sequence cut set generation, quantification, and uncertainty analysis.
C - Cut Set Editor.	Not yet implemented.
D - Display Sequences.	Calls up MENU 15, Display Sequence Data, to show cut sets, importance measures, and uncertainty results.

MENU 14, Analyze Sequences.

Analyze Sequences

Currently Selected Family —> DEMO FAMILY

Option |E| Exit / Cut Set Generation / Quantification / Uncertainty calc.

SEQUENCES

ec --> 1-LOSP-ECS	Loss of Offsite Power with success of ECS, no core da
ec --> 2-LOSP-ECS-CCS	Loss of Offsite Power, failed ECS, successful CCS, sm
ec --> 3-LOSP-ECS-CCS	Loss of Offsite Power, failed ECS, failed CCS, lg. re

Note : Changed indicators precede each sequence with meanings as follows-
e-Event Trees, c-Cut Sets, q-Quantification, u-Uncertainty
More change information may be obtained for each seq. using <Esc>.
Use <PgUp> and <PgDn> to display more sequences.

ENTERED FROM EXITS TO:

MENU 1, IRRAS 2.0 Main Menu-Sequence Analysis\ MENU 13,
Sequence Analysis System-Analyse Sequence.

OPTIONS:

E - Exit.	Returns the user to MENU 13, IRRAS-PC Sequence Analysis System.
C - Cut Set Generation.	Allows the user to select the cut set generation criteria and generate the minimal cut sets for the sequence(s) selected. The minimal cut set upper bound will be calculated also.
Q - Quantification.	Allows the user to quantify existing cut sets without regenerating them.
U - Uncertainty Calc.	Allows the user to set the uncertainty analysis parameters and conduct Monte Carlo analysis on the selected sequence cut sets.

MENU 15, Display Sequence Data.

Display Sequence Data	
Currently Selected Family —> DEMO FAMILY	
Option E	Exit / Cut Sets / Importance measures / Uncertainty data
SEQUENCES	
1-LOSP-ECS	Loss of Offsite Power with success of ECS, no core damage.
2-LOSP-ECS-CCS	Loss of Offsite Power, failed ECS, successful CCS, sm. rel.
3-LOSP-ECS-CCS	Loss of Offsite Power, failed ECS, failed CCS, lg. release.
Note : Use <PgUp> and <PgDn> to display more sequences.	

ENTERED FROM EXITS TO:

MENU 1, IRRAS 2.0 Main Menu-Sequence Analysis\ MENU 13,
Sequence Analysis System-Display Sequences

OPTIONS:

E - Exit.	Returns the user to MENU 13, IRRAS-PC Sequence Analysis System.
C - Cut sets.	Allows the user to review the quantified minimal cut sets for the selected sequence. Cut sets are sent to the screen only.
I - Importance measures.	Generates and displays the basic event importance measures for the selected sequence.
U - Uncertainty data.	Displays the uncertainty results for the selected sequence.

MENU 16, Edit Data Base Records.

Edit Data Base Records

Option |E| Exit / Family / Basic Events
 1 - Fault Trees / 2 - Event Trees & Sequences

DEMO FAMILYDemonstration sample family

Note: Select type of data to be edited, move cursor to desired family, and press <Enter>.

Use <PgUp> or <PgDn> to display more Families.

ENTERED FROM EXITS TO:

MENU 1, IRRAS 2.0 Main Menu-Modify Data Base

OPTIONS:

E - Exit.	Returns the user to MENU 1, IRRAS 2.0 Main Menu.
F - Family.	Calls up MENU 17, Family Editor, for adding, deleting, or changing a family in the data base.
B - Basic Events.	Calls up MENU 18, Edit Basic Event Data, for working with the base case basic event data.
1 - Fault Trees.	Calls up MENU 19, Edit Fault Tree, for adding, deleting, or changing fault tree data.
2 - Event Trees & Sequences.	Calls up MENU 20 for adding, deleting, or changing sequence data.

MENU 17, Family Editor.

F A M I L Y E D I T O R	
Option E	Exit / Add / Modify / Delete
Family Number	1
Family Name	DEMO FAMILY
Sequence Sum	-----E-----
Family Description	
Demonstration sample family	

Note: The Add options adds a new family record. The Modify and Delete options edit the displayed family record.

ENTERED FROM EXITS TO:

MENU 1, Main Menu-Modify Data Base\ MENU 16, Edit Data Base Records-Family.

OPTIONS:

- E - Exit. Returns the user to MENU 16, Edit Data Base Records.
- A - Add. Adds a family to the data base.
- M - Modify. Allows the user to modify data on a family in the data base.
- D - Delete. Deletes a family from the data base.

MENU 18, Edit Basic Event Data.

Edit Basic Event Data

Currently Selected Family —> DEMO FAMILY

Option | | Exit / Modify Event Data <= choose an Event
Update / Remove All unused events <= other options

C-CV-A	CCS Train A pump discharge check valve
C-CV-B	CCS Train B pump discharge check valve
C-MOV-1	CCS suction isolation valve
C-MOV-A	CCS Train A pump discharge isolation valve
C-MOV-B	CCS Train B pump discharge isolation valve
C-PUMP-A	CCS Train A motor-driven pump
C-PUMP-B	CCS Train B motor-driven pump
DG-A	Emergency diesel generator A
DG-B	Emergency diesel generator B
E-CV-A	ECS Train A pump discharge check valve
E-CV-B	ECS Train B pump discharge check valve
E-MOV-1	ECS suction isolation valve

NOTE: The Update option makes the current reliability and uncertainty values the new base case. Use <PgUp> or <PgDn> to display more Events.

ENTERED FROM EXITS TO:

MENU 1, IRRAS 2.0 Main Menu-Modify Data Base\ MENU 16,
Edit Data Base Records-Basic Events.

OPTIONS:

E - Exit.	Returns the user to MENU 16, Edit Data Base Records.
M - Modify Event Data.	Allows the user to add, delete, or change basic events and their data.
U - Update.	Updates the base case data with the current case data. The base case data is made to match the current case data.
R - Remove all unused events.	Removes all events that no longer are used in a fault tree in the current family due to data base modifications.

MENU 19, Edit Fault Tree.

Edit Fault Tree																									
Currently Selected Family —> DEMO FAMILY																									
Option	Exit / Add / Modify / Delete / Locate / Next / Previous Batch load Fault Tree logic from file																								
Fault Tree Number 1 Fault Tree Name CCS Fault Tree Level 0 (0-root trees, others-sub trees)																									
Description Containment Cooling System fault tree																									
<table border="1"><tbody><tr><td>5th %</td><td>1.542E-003</td><td>Median</td><td>9.407E-003</td><td>Min</td><td>3.669E-004</td><td>Std. Dev</td><td>3.776E-002</td></tr><tr><td>95th %</td><td>7.255E-002</td><td>Mean</td><td>2.061E-002</td><td>Max</td><td>3.933E-001</td><td>Skewness</td><td>5.385E+000</td></tr><tr><td colspan="2">Min Cut Upper Bound</td><td colspan="2">2.120E-002</td><td colspan="2">Kurtosis</td><td colspan="2">4.030E+001</td></tr></tbody></table>		5th %	1.542E-003	Median	9.407E-003	Min	3.669E-004	Std. Dev	3.776E-002	95th %	7.255E-002	Mean	2.061E-002	Max	3.933E-001	Skewness	5.385E+000	Min Cut Upper Bound		2.120E-002		Kurtosis		4.030E+001	
5th %	1.542E-003	Median	9.407E-003	Min	3.669E-004	Std. Dev	3.776E-002																		
95th %	7.255E-002	Mean	2.061E-002	Max	3.933E-001	Skewness	5.385E+000																		
Min Cut Upper Bound		2.120E-002		Kurtosis		4.030E+001																			

ENTERED FROM EXITS TO:

MENU 1, IRRAS 2.0 Main Menu-Modify Data Base\ MENU 16,
Edit Data Base Records-1-Fault Trees

OPTIONS:

E - Exit.	Returns the user to MENU 16, Edit Data Base Records.
A - Add.	Adds a fault tree to the data base.
M - Modify.	Allows the user to modify data on a fault tree in the data base.
D - Delete.	Deletes a fault tree from the data base.
L - Locate.	Searches for the designated fault tree data.
N - Next.	Calls up the next fault tree in the data base.
P - Previous.	Calls up the fault tree in the data base listed just above the fault tree shown.
B - Batch load Fault Tree Logic from File.	Loads all fault tree logics existing in the current directory in the form of .TRE files into the data base.

MENU 20, Event Tree Editor.

Event Tree Editor

Currently Selected Family —> DEMO FAMILY

Option | | Exit / Add / Modify / Delete / Locate / Next / Previous
Sequence Edit

Event Tree Number 1

Event Tree Name LOSP

Event Tree Description
Loss of Offsite Power event tree.

Note: The Add options adds a new event tree record. The Modify and Delete options edit the displayed event tree record. The Sequence Edit option allows editing of associated sequences.

ENTERED FROM EXITS TO:

MENU 1, IRRAS 2.0 Main Menu-Modify Data Base\ MENU 16,
Edit Data Base Records-2-Event Tree Sequences

OPTIONS:

E - Exit.	Returns the user to MENU 16, Edit Data Base Records.
A - Add.	Adds an event tree to the data base.
M - Modify.	Allows the user to modify data on an event tree in the data base.
D - Delete.	Deletes an event tree from the data base.
L - Locate.	Searches for the designated event tree data.
N - Next.	Calls up the next event tree in the data base.
P - Previous.	Calls up the event tree in the data base listed just above the event tree shown.
S - Sequence Edit.	Calls up MENU 21, Edit Sequence, for access to the sequence data base.

MENU 21, Edit Sequence.

Edit Sequence							
Option			Exit / Add / Modify / Delete / Locate / Next / Previous				
Include Fault Trees in Sequence							
Family	DEMO	FAMILY	Event Tree	LOSP	Sequence	1-LOSP-ECS	
Description	Loss of Offsite Power with success of ECS, no core damage.						
5th %	9.267E-002	Median	7.969E-001	Min	5.228E-003	Std. Dev	6.782E+000
95th %	8.676E+000	Mean	2.508E+000	Max	1.117E+002	Skewness	9.037E+000
	Min Cut	Upper Bound	2.300E+000			Kurtosis	1.121E+002

ENTERED FROM EXITS TO:

MENU 1, IRRAS 2.0 Main Menu-Modify Data Base\ MENU 16,
Edit Data Base Records-2-Event Tree Sequence\
MENU 20, Event Tree Editor-Sequence Edit.

OPTIONS:

E - Exit.	Returns the user to MENU 20, Event Tree Editor.
A - Add.	Adds a sequence to the event tree.
M - Modify.	Allows the user to modify data on a sequence in the data base.
D - Delete.	Deletes a sequence from the data base.
L - Locate.	Searches for the designated sequence data.
N - Next.	Calls up the next sequence in the data base.
P - Previous.	Calls up the sequence in the data base listed just above the sequence shown.
I - Include Fault Trees in Sequence.	Calls up MENU 22, Include Fault Trees, to designate the logic of the sequence.

MENU 22, Include Fault Trees.

I n c l u d e F a u l t T r e e s

Option |1| Exit / Include Fault Tree in Sequence

Family DEMO FAMILY Event Tree LOSP Sequence 1-LOSP-ECS
 Description Loss of Offsite Power with success of ECS, no core damage.

CCS	ECS	LOSP	

Note: To include a fault tree () or its compliment () or remove it from the sequence, use <I> option. Use <PgUp> or <PgDn> to display more trees.

ENTERED FROM EXITS TO:

MENU 1, IRRAS 2.0 Main Menu-Modify Data Base\ MENU 16, Edit Data Base-2-Event Tree Sequence\ MENU 20, Event Tree Editor-Sequence Edit\ MENU 21, Edit Sequences-Include Fault Tree in Sequence.

OPTIONS:

<p>E - Exit.</p> <p>I - Include Fault Tree in Sequence.</p>	<p>Returns the user to MENU 21, Include Fault Trees.</p> <p>Allows the user to define the logic of a sequence in terms of the successful and failed fault tree top events.</p>
---	--

MENU 23, IRRAS Reports.

```

      I R R A S   R E P O R T S

Option |E| Exit / Basic Event / Fault Tree / Sequence    <= choose a family
        Report of Families                             <= other options

DEMO FAMILY          Demonstration sample family


Note: Select desired report, press <Enter> to continue.
      Use <PgUp> or <PgDn> to see more Family names.
```

ENTERED FROM EXITS TO:

MENU 1, IRRAS 2.0 Main Menu-Report on Data Base

OPTIONS:

E - Exit.	Returns the user to MENU 1, IRRAS 2.0 Main Menu.
B - Basic Event.	Calls up MENU 24, Basic Event Reports, for access to basic event data reports.
F - Fault Tree.	Calls up MENU 25, Fault Tree Reports, for access to fault tree data reports.
S - Sequence.	Calls up MENU 26, Sequence Reports, for access to sequence data reports.
R - Report of Families.	Generates the Family Summary Report (Appendix A, Report 17).

MENU 24, Basic Event Reports.

<table border="1"><tr><td>IRRAS REPORTS</td></tr><tr><td>Basic Event Reports</td></tr></table>	IRRAS REPORTS	Basic Event Reports
IRRAS REPORTS		
Basic Event Reports		
Currently Selected Family => DEMO FAMILY		
Option... E Exit / Summary / Probabilities / Uncertainty Values		

ENTERED FROM EXITS TO:

MENU 1, IRRAS 2.0 Main Menu-Report on Data Base\ MENU 23,
IRRAS Reports-Basic Events.

OPTIONS:

E - Exit.	Returns the user to MENU 23, IRRAS Reports.
S - Summary.	Generates the Basic Events Summary Report
(Appendix A, Report 1).
P - Probabilities.	Generates the Basic Events Probability Report
(Appendix A, Report 2).
U - Uncertainty
Values.	Generates the Basic Events Uncertainty Values
Report (Appendix A, Report 3).

MENU 25, Fault Tree Reports.

<div style="border: 1px solid black; display: inline-block; padding: 2px 10px;">IRRAS REPORTS Fault Tree Reports</div>									
Currently Selected Family => DEMO FAMILY									
Option E Exit / Logic / eXpanded Logic / Modified Logic / Importance Measures / Cut Sets / Summary / Uncertainty Values All Levels / Zero Levels									
<table style="width: 100%; border: 1px solid black;"><thead><tr><th style="width: 20%; text-align: left; padding: 2px;">FAULT</th><th style="width: 80%; text-align: left; padding: 2px;">TREES</th></tr></thead><tbody><tr><td style="padding: 2px;">CCS</td><td style="padding: 2px;">Containment Cooling System fault tree</td></tr><tr><td style="padding: 2px;">ECS</td><td style="padding: 2px;">Emergency Cooling System fault tree</td></tr><tr><td style="padding: 2px;">LOSP</td><td style="padding: 2px;">Loss of Offsite Power Initiating Event fault tree</td></tr></tbody></table>		FAULT	TREES	CCS	Containment Cooling System fault tree	ECS	Emergency Cooling System fault tree	LOSP	Loss of Offsite Power Initiating Event fault tree
FAULT	TREES								
CCS	Containment Cooling System fault tree								
ECS	Emergency Cooling System fault tree								
LOSP	Loss of Offsite Power Initiating Event fault tree								
Note : Use <PgUp> and <PgDn> to display more fault trees. To execute options (L,X,M,I,C) you must highlight a fault tree.									

ENTERED FROM EXITS TO:

MENU 1, IRRAS 2.0 Main Menu-Report on Data Base\ MENU 23,
IRRAS Reports-Fault Tree

OPTIONS:

E - Exit.	Returns the user to MENU 23, IRRAS Reports.
L - Logic.	Generates the Fault Tree Logic Report (Appendix A, Report 4).
M - Modified Logic.	Generates the Fault Tree Modified Logic Report (Appendix A, Report 5).
I - Importance Measures.	Generates the Fault Tree Importance Measures Report (Appendix A, Report 6).
C - Cut Sets.	Generates the Fault Tree Cut Sets Report (Appendix A, Report 7) and the Fault Tree Cut Sets (Quantification) Report (Appendix A, Report 8).
S - Summary.	Generates the Fault Tree Summary Report (Appendix A, Report 9).
U - Uncertainty Values.	Generates the Fault Tree Uncertainty Values Report (Appendix A, Report 10).

MENU 26, Sequence Reports.

I R R A S R E P O R T S							
Sequence Reports							
Currently Selected Family => DEMO FAMILY							
Option E	Exit / Logic / Summary / Uncertainty Values <= other options Importance Measures / Cut Sets <= choose a sequence						
<div><div>SEQUENCES</div><table><tbody><tr><td>1-LOSP-ECS</td><td>Loss of Offsite Power with success of ECS, no core damage.</td></tr><tr><td>2-LOSP-ECS-CCS</td><td>Loss of Offsite Power, failed ECS, successful CCS, sm. rel.</td></tr><tr><td>3-LOSP-ECS-CCS</td><td>Loss of Offsite Power, failed ECS, failed CCS, lg. release.</td></tr></tbody></table></div>		1-LOSP-ECS	Loss of Offsite Power with success of ECS, no core damage.	2-LOSP-ECS-CCS	Loss of Offsite Power, failed ECS, successful CCS, sm. rel.	3-LOSP-ECS-CCS	Loss of Offsite Power, failed ECS, failed CCS, lg. release.
1-LOSP-ECS	Loss of Offsite Power with success of ECS, no core damage.						
2-LOSP-ECS-CCS	Loss of Offsite Power, failed ECS, successful CCS, sm. rel.						
3-LOSP-ECS-CCS	Loss of Offsite Power, failed ECS, failed CCS, lg. release.						
Note : Use <PgUp> and <PgDn> to display more sequences.							

ENTERED FROM EXITS TO:

MENU 1, IRRAS 2.0 Main Menu-Report on Data Base\ MENU 23,
IRRAS Reports-Sequence.

OPTIONS:

E - Exit.	Returns the user to MENU 23, IRRAS Reports.
L - Logic.	Generates the Sequence Logic Report (Appendix A, Report 11).
S - Summary.	Generates the Sequence Summary Report (Appendix A, Report 12).
U - Uncertainty Values.	Generates the Sequence Uncertainty Values Report (Appendix A, Report 13).
I - Importance Measures.	Generates the Sequence Importance Measures Report (Appendix A, Report 14).
C - Cut Sets.	Generates the Sequence Cut Sets Report (Appendix A, Report 15) and the Sequence Cut Sets (Quantification) Report (Appendix A, Report 16).

MENU 27, Utility Options.

<table border="1"><tr><td>IRRAS - P C Utilities</td></tr></table> <table border="1"><tr><td>E - Exit D - Define Constants C - Communications P - PRA Data Base Interface I - IRRAS/SETS Interface V - Version 1.0/2.0 Interface R - Recover DataBase</td></tr></table> Option... E	IRRAS - P C Utilities	E - Exit D - Define Constants C - Communications P - PRA Data Base Interface I - IRRAS/SETS Interface V - Version 1.0/2.0 Interface R - Recover DataBase
IRRAS - P C Utilities		
E - Exit D - Define Constants C - Communications P - PRA Data Base Interface I - IRRAS/SETS Interface V - Version 1.0/2.0 Interface R - Recover DataBase		

ENTERED FROM EXITS TO:

MENU 1, IRRAS 2.0 Main Menu-Utility Option.

OPTIONS:

E - Exit.	Returns the user to MENU 1, IRRAS 2.0 Main Menu.
D - Define Constants.	Allows the user to set defaults for system operation (See Section 3.1).
C - Communications.	Initiates the CONVERSE terminal emulation communications processor for mainframe communications.
P - PRA Data Base Interface	Not yet implemented
I - IRRAS/SETS Interface.	Calls up MENU 28, IRRAS to SETS Interface, for converting data from one format to the other.
V - Version 1.0/2.0 Interface.	Calls up MENU 29, Version 1.0/2.0 Interface, for converting from one IRRAS format to the other.
R - Recover Data Base.	Restructures and organizes the data base after extensive use and editing or after a program error.

MENU 28, IRRAS to SETS Interface.

ENTERED FROM EXITS TO:

MENU 1, IRRAS 2.0 Main Menu-Utility Options\ MENU 27,
Utility Option-IRRAS/SETS Interface.

OPTIONS:

E - Exit.	Returns the user to MENU 27, Utility Options.
S - Select Family.	Calls up MENU 3, Select Family of Fault Trees.
I - IRRAS to SETS.	Converts IRRAS 2.0 logics to SETS fault tree input files.
X - SETS to IRRAS.	Converts SETS input files to IRRAS 2.0 logic files.

MENU 29, Version 1.0/2.0 Interface.

ENTERED FROM EXITS TO:

MENU 1, IRRAS 2.0 Main Menu-Utility Option\ MENU 27,
Utility Option-Version 1.0\2.0 Interface.

OPTIONS:

E - Exit.	Returns the user to MENU 27, Utility Options.
S - Select Family.	Calls up MENU 3, Select Family of Fault Trees.
1 - Version 1 to 2.	Allows the user to convert IRRAS 1.0 data to IRRAS 2.0 format.
2 - Version 2 to 1.	Allows the user to convert IRRAS 2.0 data to IRRAS 1.0 format.

APPENDIX E

CONVERSE COMMANDS

APPENDIX E
CONVERSE COMMANDS

System Parameters

Table E-1 lists the system parameters in CONVERSE.

SYSTEM PARAMETERS		
F1	Baud Rate	1200
F2	Stop Bits	1
F3	Parity Bit	OFF
F4	Parity	ODD
F5	Number of Bits	8
F6	Xon/Xoff Flow Control	ON
F7	Echo	ON
F8	Screen Save Option	ON
F9	Auto Line Feed	OFF
F10	Communications Port	COM1
Depress F1-F10 keys to change parameter settings,<CR> to exit		

Table E-1. CONVERSE System Parameters.

Baud rate is the rate at which CONVERSE transmits and receives data. By depressing the F1 (function 1) key, the various values from 300 to 19200 will be presented.

The number of stop bits may be set at 1 or 2 by depressing the F2 key. The default is 1.

The parity bit setting determines whether parity is used or not. It is set to OFF (default) or ON by depressing the F3 key. If the parity bit is ON, then the parity setting (F4) determines whether the parity is ODD (default) or EVEN.

The number of bits setting (F5) is the number of valid bits used per character. The default is 8. It may be a value from 5 to 8.

Flow control (F6) determines whether flow control is used or not. If it is used (the default), then the DC1 (XON) and DC3 (XOFF) ASCII characters are used to control the flow of information.

Echo (F7) ON causes each character typed at the keyboard to be echoed to the screen. The default is ON.

The Screen Save Option (F8) causes all data sent or typed to the screen currently held in screen memory to be saved to the file named CONVERSE.SAV in the main or root directory when the program is terminated. The screen scrollable memory may contain up to 375 lines of data. If the entire screen memory is blank, the CONVERSE.SAV file will be deleted. It may be restored to the screen at a later time using the Ctrl/F3 keys. The default is ON.

Auto Line Feed (F9) causes CONVERSE to automatically generate a line feed (LF) after it receives a carriage return character (CR) from the host. The default is OFF.

The Communications Port (F10) option allows selection of the communications port on your system through which CONVERSE will communicate with the host. Ports 1 and 2 are available. The default is port 1.

CONVERSE Options

CONVERSE is used to provide VT100 terminal emulation capabilities within a program. This program attempts access to a file called CONVERSE.SET which holds its communications settings such as baud rate, parity, etc. If this file does not exist, the default settings will be shown to the user for changes, and then the file will be created. The settings may be changed at any time from within the CONVERSE procedure by the user when the Ctrl/F2 keys are depressed. The four arrow keys cause cursor movement in their customary directions. The keys in Table E-2 are also provided as capabilities within the system. (Alt/H depressed by the user will show this help table on the screen.)

Key(s)	Description	Key(s)	Description
^->	Cursor right 1 word	^<-	Cursor left 1 word
Home	Cursor and page home	End	End of current line
PgUp	Page up a screen	PgDn	Page down a screen
Ins	Insert a character	Del	Delete a character
Alt/A	Insert a line after	Alt/B	Insert a line before
Alt/D	Delete current line	Alt/I	Insert mode on/off
Alt/n	(n=1..5) store line	Alt/n	(n=6..0) retrieve ln
Shft/F1	Clear Screen	Shft/F2	Insert a line after
Shft/F3	Printer on/off	Shft/F4	COMMAND MODE
Shft/F5	Capture to file on/off	Shft/F6	Restore from file
Shft/F7	Foreground color	Shft/F8	Background color
Shft/F9	Online/offline	Shft/F10	Send current line
Ctrl/F1	Reset system	Ctrl/F2	System definition
Ctrl/F3	Restore saved screen		
^Z	Exit Converse	^PGDN	Erase - cursor to end
^End	Erase - cur. to end line	^Home	Clear Screen

Table E-2. CONVERSE Key Descriptions.

Commands

The COMMAND option (invoked online by the F4 key) allows the execution of a CONVERSE command. This may be a file (.CMD file type) or a predefined command. As may be noticed, a command such as is entered here may also be invoked as the CONVERSE PROCEDURE is first executed by including a command in the argument list. In this way, CONVERSE may be instructed to execute a series of commands for the purpose of talking through a modem, connecting to a mainframe system, etc. CONVERSE commands include the following:

```
filename  AWAIT  BELL  CAPTURE  CLEAN  CLEAR  DIAL  DISPLAY
EXIT      HIDE  IF    PAUSE    PREVIOUS PRINT REQUEST RESET
RESTORE   SEND  SET   STOP     TIMER
```

A command may be invoked directly by the user typing it in, it may be part of a command file, or it may be conditionally executed from other commands such as AWAIT, TIMER, or PREVIOUS.

Strings

Many commands use a string. A string may be defined in several ways and composed of many parts. The (") and the (')

characters may be used in pairs to show string contents. The CHR(n) designation may be used to specify an ASCII character by number (where the n = character #). A semicolon (;) outside of a pair of quotes represents a carriage return (cr) character (CHR(13)). A comma may be used to separate parts of a string. A predefined variable (such as from REQUEST command) may be used in the string. For example:

```
REQUEST "Please enter a value .." abc
SEND 'this is a string ',CHR(10),example,abc;
```

The string defined in the previously given SEND command example is composed of 5 parts which are:

- 1) The string in quotes ('this is a string').
- 2) The ASCII character 10 (line feed).
- 3) The string, "example" (notice that it need not be in quotes).
- 4) The contents of the "abc" variable as entered by the user when the previous REQUEST command was executed.
- 5) <cr> (carriage return character) represented by the ";;".

Defined Commands

- o AWAIT. Wait for one or more strings to be sent by the host before executing the next command. If a single string is being awaited, the command need not be specified since the next command in the file will be executed. If awaiting multiple strings, the first string encountered will cause its companion command to be executed. If that command is a blank string or non-existent command, the next command following the AWAIT will be executed. Up to 3 different string/command pairs may be specified. If only 1 or 2 pairs are needed, the remaining pairs need not be specified.

format:

```
AWAIT string command string command string command
```

examples:

```
AWAIT "CONNECT" " " "NO CARRIER" STOP
AWAIT 'my string',abc;
```

- o CLEAR. Clear the screen and position cursor at the top.
format:
CLEAR
- o BELL. Ring the Bell.
format:
BELL
- o CAPTURE. Capture all information that is sent to the screen to a specified file. If already capturing information to a file when this command is issued, then the previous file used for capturing is closed. The second execution of the CAPTURE command therefore acts as a toggle to close the file specified in the prior CAPTURE command.
format:
CAPTURE filename
CAPTURE
- o CLEAN. Clean out all commands which follow this command so that they will not be executed.
format:
CLEAN
- o DISPLAY. Display a message at the bottom of the screen on line 25.
format:
DISPLAY string
- o EXIT. Leave the CONVERSE PROCEDURE without clearing the screen or terminating communications. CONVERSE may then be reentered and continue with the communications later.
format:
EXIT

- o HIDE. Cause the keyboard or host characters to not be displayed on the screen. The HIDE KEYBOARD command also effects any entry by the user with the REQUEST command. For this type of entry, each keystroke causes an asterisk (*) to appear on line 25.

format:

HIDE KEYBOARD/HOST/NONE

- o IF. Check for the status of a flag. Up to 30 different status flags may be set. This statement allows the checking of a flag along with the execution of any subsequent commands.

format:

IF flag command
IF NOT flag command

examples:

IF NOT FLAG3 "SEND ERROR;"
IF FLAG2 PRINT

- o PAUSE. Wait for the prescribed number of seconds. If characters are being sent from a host, they will still be received but not processed (displayed, etc.) until the pause is over and control returns from the PAUSE command.

format:

PAUSE (# seconds)

- o PREVIOUS. Execute a command based on the characters preceding the cursor in its current position on the screen. This command is useful if you are unsure of what actions have just taken place. By investigating the prompt or other characters immediately following the cursor on the same line, you may be able to determine which functions to perform. It is similar to the AWAIT command in construct and function except that the string has already been sent.

format:

PREVIOUS string command string command string command

example:

PREVIOUS "Ready ... " " " Error "SEND 'An error has occurred'"

- o PRINT. Toggle the printer ON or OFF. If the printer is already OFF, then issuing this command will cause it to be turned ON. When printing, all information which is sent to be displayed on the screen will also be sent to the printer. If the printer is currently in the ON state and this command is issued, then it is turned OFF.

format:

PRINT

- o REQUEST. Request entry from the user and store in a variable. Any variable name up to 8 characters in length may be used. Upper and lower case are not recognized. A maximum of 10 different variables may be defined during the course of CONVERSE execution. The defined variable may be used later as part of a string.

format:

REQUEST string variable

- o RESET. Reset the communications port. This is necessary to reestablish communications with new parameters if previously defined using the SET command. It is not necessary when first entering CONVERSE, since communications are automatically established during this process.

format:

RESET

- o RESTORE. Read the contents of the specified disk file and send it simultaneously to the screen (if local echo is ON) and to the host.

format:

RESTORE filename

- o SEND. Send a string of characters to the host through a connected port.

format:

SEND string

example:

```
SEND 'hello';
```

- o SET. Set CONVERSE parameters. Some parameters are for communication and do not take effect until CONVERSE is reentered or the RESET command is issued to reestablish communications. These type include BAUD, STOPBITS, PARITYBIT, PARITY, BITSPERCHAR, FLOW (XON/XOFF flow control), and COMPORT (communications port number).

Other commands are ECHO (local echo), SCREENSAVE (save the screen in scrollable memory to the CONVERSE.SAV file upon leaving), AUTOLF (generate a LF character after each CR is received), COLOR (set the foreground and background colors), and STOPCHAR (specify the character used to terminate CONVERSE - default is ^z).

- o FLAG. The FLAG command allows the setting of one of the 30 available status flags in CONVERSE. The first flag is named FLAG1, the second FLAG2, and so forth. Each flag may be set to TRUE (ON), or FALSE (OFF) or reversed from its current setting.

format:

SET	BAUD	300/600/1200/2400/4800/9600/19200
	STOPBITS	0/1/2
	PARITYBIT	ON/OFF
	PARITY	EVEN/ODD
	BITSPERCHAR	5/6/7/8
	FLOW	ON/OFF
	COMPORT	1/2
	ECHO	ON/OFF
	SCREENSAVE	ON/OFF
	AUTOLF	ON/OFF
	COLOR	(color number -> blink-R-G-B-Intense-R-G-B)
	STOPCHAR	(char. number)
	FLAGn	ON/TRUE/OFF/FALSE/REVERSE

examples:

```
SET BAUD 9600
SET FLAG21 REVERSE
SET SCREENSAVE OFF
SET COLOR 18
SET STOPCHAR 26
```

- o STOP. Terminate communications and leave the CONVERSE PROCEDURE.

format:

STOP

- o TIMER. Await for defined strings from the host for only a prescribed length of time. This command is also used in similar fashion to the PAUSE command except that, like the AWAIT command, processing and displaying of interactions with the host and user is performed during the timed period. This command differs from AWAIT because it will only wait for the prescribed time. If one of the strings being awaited is not received in this period of time, the command following the specified number of seconds will be executed instead. If awaiting multiple strings, the first string encountered will cause its companion command to be executed. If that command is a blank string or non-existent command, the next command following the AWAIT will be executed. Up to 2 different string/command pairs may be specified following the (# of sec.)/command pair. If only 1 pair is needed, the remaining pair need not be specified.

format:

TIMER (# of sec.) command string command string command
TIMER (# of sec.)

examples:

TIMER 10 STOP "Ready ..." "SEND RETURN" Error " "
TIMER 5

APPENDIX F
DATA INTERCHANGE FORMATS

MAR - D
PRA Models and Results Data Base
Data Interchange Formats. March 22, 1989

Family (Plant) Information

Family Names and Descriptions

File Name:
xxxxxxx.FAD

File Format:
name,description

Where:
name - 16 character Family name.
(First 8 characters must unique).
description - 60 character Family description

Family Attribute File

File Name:
xxxxxxx.FAA

File Format:
name,mission,newSum,co,loc,type,design,vendor,AE,OpDate,QDate

where:

name	- 16 character	Family name.
mission	- Floating point	Default mission time in hours.
newSum	- Floating point	New sequence frequency sum.
co	- 10 character	Company name.
loc	- 16 character	Location name.
type	- 3 character	Facility type.
design	- 10 character	Facility design.
vendor	- 5 character	Vendor name.
AE	- 10 character	Architectural Engineer.
OpDate	- (yyyy/mm/dd)	Operational date
QDate	- (yyyy/mm/dd)	Qualification date

Family Textual Information

File Name:
xxxxxx.FAT

File Format:
family =
-- text --

where:
family - 16 character Family name

General Rules:

1. File is standard ASCII text. Single spaced. Upper and lower case.
2. Start of paragraph indented 5 spaces. Blank line between paragraphs.

Basic Event Information

Basic Event Names and Descriptions

File Name:
xxxxxx.BED

File Format:
family =
name,description
... , ...

where:

family	- 16 character	Family name
name	- 16 character	Event name
description	- 60 character	Alphanumeric description

Basic Event Rate Information

File Name:
xxxxxx.BEI

File Format:
family =
name, calc, udC, udT, udV, prob, lamda, tau, mission, init
... , ... , ..., ..., ..., ... , ... , ..., ... , ...

where:

family	- 16 character	family name
name	- 16 character	Basic event name
calc	- 1 character	Calculation type

1	- Probability
2	- Lambda * Mission Time
3	- $1 - \text{Exp}(-\text{Lambda} * \text{Mission Time})$
4	- $\text{Lambda} * \text{Min}(\text{Mission Time}, \text{Tau})$
5	- Operating component with full repair
6	- $\text{Lambda} * \text{Tau} / 2.0$
7	- $1 + (\text{EXP}(-\text{Lambda} * \text{Tau}) - 1.0) / (\text{Lambda} * \text{Tau})$
8	- Base Probability * Probability
9	- Base Probability * Probability
T	- Set to House Event (Failed, Prob=1.0)
F	- Set to House Event (Successful, Prob=0.0)

udC	- 3 digit integer	Uncertainty correlation class Events in same class are 100% correlated.
udT	- 1 character	Uncertainty distribution type

udV	- floating point	Uncertainty distribution value
prob	- floating point	Probability value
lambda	- floating point	Basic event failure rate per hr.
tau	- floating point	Time to repair in hours
mission	- floating point	Mission time
init	- boolean (T/F)	Initiating event

Rules:

1. The "name" field is mandatory.
2. If a field is omitted insert the "," as a place holder.
3. Any number of spaces may be inserted between fields.

Basic Event Attribute Codes

File Name:

xxxxxx.BEA

File Format:

family =
name,Aname,type,sys,fail,loc,compID,Gname,train,att1,...,att16
..., ... ,... ,...,... ,..., ... , ... , ... ,... ,...,...

where:

family	- 16 character	Family name
name	- 16 character	Event name
Aname	- 16 character	Alternate event name
type	- 3 character	Event component type
sys	- 3 character	Event component system
fail	- 2 character	Failure mode
loc	- 3 character	Component location
compID	- 7 character	Component ID
Gname	- 16 character	Event group identifier
train	- 1 character	Train identifier
att1 ... att16	- class attribute flags	
		16 values of Y or N (for yes or no)
		indicates whether the attribute described
		in the "Class Attribute" file is applicable

Rules:

1. The "name" field is mandatory.
2. If a field is omitted insert the "," as a place holder.
3. Any number of spaces may be inserted between fields.

Event Attribute Descriptions

Failure Mode Descriptions

File Name:
xxxxxx.FMD

File Format:
family =
fail,description
... ,...

where:

family	- 16 character	Family name
fail	- 2 character	Failure mode identifier
description	- 60 character	Failure mode description

Component Type Descriptions

File Name:
xxxxxx.CTD

File Format:
family =
comp, description
... ,...

where:

family	- 16 character	Family name
comp	- 3 character	Component type identifier
description	- 60 character	Component type description

System Type Descriptions

File Name:
xxxxxx.STD

File Format:
family =
sys,description
... ,...

where:

family	- 16 character	Family name
sys	- 3 character	Component system identifier
description	- 60 character	System description

Location Descriptions

File Name:
xxxxxx.LCD

File Format:
family =
loc,description
... ,...

where:

family	- 16 character	Family name
loc	- 3 character	Component location identifier
description	- 60 character	Component location description

Class Attribute Descriptions

File Name:
xxxxxx.CAD

File Format:
family =
Attr#,description
... ,...

where:

family	- 16 character	Family name
Attr#	- integer 1..16	Attribute number
description	- 60 character	Attribute description

Fault Tree Information

Fault Tree Names and Descriptions

File Name:
xxxxxxx.FTD

File Format:
family =
name,description
... ,...

where:

family	- 16 character	Family name
name	- 16 character	Fault tree name
description	- 60 character	Fault tree description

Fault Tree Graphics

File Name:
xxxxxxx.DLS

File Format:
IRRAS 2.0 Fault Tree Graphics file (DLS format).

Fault Tree Logic

File Name:

xxxxxx.FTL

File Format:

```
family, fault tree =  
(logic output from IRRAS 2.0 in IRRAS 1.0 format.)  
* gatenamel,description  
gatenamel gatetype input1 input2 ... inputn  
...      ...      ...      ...  
* gatenamen,description  
gatenamen gatetype input1 input2 ... inputn
```

where:

family	- 16 character	Family name
fault tree	- 16 character	Fault tree name
gatenamel	- 16 character	Gate name
gatetype	- 4 character	Gate type

BE = basic event
AND = logical AND
OR = logical OR
TBL = table of events
TRAN = transfer - 16 character name
following is a fault tree name
NAND = logical NOT AND
NOR = logic NOT OR
N/M = N out of M logic gate
CONT = continuation gate

input	- 16 character	inputs to the gate (event or gate names)
description	- 60 character	gate name descriptions included as comments

General Rules:

1. No gate definition may exceed 255 characters.
2. No line starting with an asterisk ("*") is a comment.
3. For each gate name a comment should be included in the format shown to give the gate description.

Fault Tree Cut Sets

File Name:
xxxxx.FTC

File Format:
family, fault tree =
eventname * eventname +
eventname * eventname * eventname *
eventname +
eventname * eventname.

where:

family	- 16 character	Family name
fault tree	- 16 character	Fault tree name
eventname	- 16 character	Event names in the cut set.

General Rules:

1. Cut set events are separated by asterisks ("*"). Spaces are ignored.
2. Cut sets are separated by a plus ("+").
3. A period (".") denotes the end of the sequence.
4. Complimented events are preceded by a slash ("/").
5. Event names are a maximum of 16 characters including the "/" character for complimented events.
6. A line starting with an asterisk ("*") is a comment.

Fault Tree Attributes

File Name:
xxxxx.FTA

File Format:
family =
name, level, mission, mincut, proCut, sample, seed, sizCut, sys, cuts,
events, value1, ..., value9
... , ... , ... , ... , ... , ... , ... , ... , ... ,

where:

family	- 16 character	Family name
name	- 16 character	Fault tree name
level	- Integer 2	0 = top level tree
mission	- Floating point	Mission time
mincut	- Floating point	Mincut upper bound
proCut	- Floating point	Probability cut off value
sample	- Integer 4	Sample size
seed	- Integer 8	Random number seed
sizecut	- Integer 2	Size cut off value
sys	- 3 character	System identifier
cuts	- Integer 5	Base number of cutsets
events	- Integer 5	Base number of events

value - Floating point Base uncertainty values

General Rules:

1. If a piece of information is not available the comma (",") should be included as a placeholder.
2. Extra blanks between fields are ignored.

Fault Tree Textual Information

File Name:

xxxxxx.FTT

File Format:

family, fault tree =
-- text --

where:

family	- 16 character	Family name
fault tree	- 16 character	Fault tree name

General Rules:

1. File is standard ASCII text.
Single spaced.
Start of paragraph indented 5 spaces.
Blank line between paragraphs.
Upper and lower case.

Event Tree Information

Event Tree Names and Descriptions

File Name:
xxxxxx.ETD

File Format:
family =
name,description
... ,...

where:

family	- 16 character	Family name
name	- 16 character	Event tree name
description	- 60 character	Event tree description

Event Tree Attributes

File Name:
xxxxxx.ETA

File Format:
family =
name,init
... ,...

where:

family	- 16 character	Family name
name	- 16 character	Event tree name
init	- 16 character	Initiating event name

Event Tree Graphics

File Name:
xxxxxx.ETG

File Format:
IRRAS 2.0 Event Tree Graphics file.

IRRAS SAMPLE GRAPHICAL EVENT TREE

Event Tree Logic

File Name:

xxxxxx.ETL

File Format:

family, event tree, init event =

^TOPS

* 1 | 2 | 3 | 4 | 5 | this is a comment

ABCDE BCDEF CDEFG DEFGH EFGHI

^LOGIC

```

+1      +2      3      +4      +5
                        -5
                        -4      5
-2      +3      4      5
        -3      +4      +5
                        -5
                        -4      5
  
```

-1	+2	3	4	5
	-2	+3	+4	5
			-4	5
			-4	5
			-4	5
	-3		4	5

^SEQUENCES

Y/N, header#1 ,Y/N, header#2, Y/N, header#3, Y/N,header#4
Y/N, sequence#1 ,Y/N, end state#1, Y/N, xdata1#1, Y/N,xdata2#1
Y/N, sequence#2 ,Y/N, end state#2, Y/N, xdata1#2, Y/N,xdata2#2
Y/N, sequence#3 ,Y/N, end state#3, Y/N, xdata1#3, Y/N,xdata2#3
Y/N, sequence#4 ,Y/N, end state#4, Y/N, xdata1#4, Y/N,xdata2#4
Y/N, sequence#5 ,Y/N, end state#5, Y/N, xdata1#5, Y/N,xdata2#5
Y/N, sequence#6 ,Y/N, end state#6, Y/N, xdata1#6, Y/N,xdata2#6
Y/N, sequence#7 ,Y/N, end state#7, Y/N, xdata1#7, Y/N,xdata2#7
Y/N, sequence#8 ,Y/N, end state#8, Y/N, xdata1#8, Y/N,xdata2#8
Y/N, sequence#9 ,Y/N, tran file#9, Y/N, xdata1#9, Y/N,xdata2#9, T
Y/N, sequence#10,Y/N, end state#10,Y/N, xdata1#10,Y/N,xdata2#10
Y/N, sequence#11,Y/N, end state#11,Y/N, xdata1#11,Y/N,xdata2#11
Y/N, sequence#12,Y/N, end state#12,Y/N, xdata1#12,Y/N,xdata2#12
Y/N, sequence#13,Y/N, end state#13,Y/N, xdata1#13,Y/N,xdata2#13

^TEXT

SIZE s
JUST j
COLOR j
XY xvalue,yvalue
"60 character line of text"
XY xvalue, yvalue
"60 character line of text"
"60 character line of text"

^PARMS

START yvalue
WINDOW x1,y1,x2,y2

where:

family	- 16 character	Family name
event tree	- 16 character	Event tree name
init event	- 16 character	Initiating Event
TOPS	- 16 character	Top event/system names
Y/N	-	Endstate text displayed? Yes/No
header	- 16 character	Sequence header
sequence	- 16 character	Sequence name
endstate	- 16 character	End state name
tran file	- 16 character	Name of transfer file
xdata1	- 16 character	Information (optional)
xdata2	- 16 character	Information (optional)

General Rules:

1. "*" at start of line represents a comment.
2. Literal "^TOPS", "^LOGIC", "^SEQUENCES" labels must be present.
3. Logic is built according to the position of the top event in the definition.
 A "+" indicates that the specified top event was successful.
 A "-" indicates that the specified top event failed.
 A " " indicates that the response of the indicated top event did not matter.
4. Header, Sequence name, End State name, Xdata1, Xdata fields associated with each sequence. "Y/N" is a flag indicating whether the specified field is visible or not. A "T" at the end indicates the sequence transfers to another tree.
5. User text is input following the ^TEXT command. Parameters include the size, justification, color, and location of the text block.
6. The ^PARMS command allows input of program control parameters.

Event Tree Textual Information

File Name:

xxxxxx.ETT

File Format:

family, event tree =
 -- text --

where:

family	- 16 character	Family name
event tree	- 16 character	Event tree name

General Rules:

1. File is standard ASCII text.
 Single spaced.
 Start of paragraph indented 5 spaces.
 Blank line between paragraphs.
 Upper and lower case.

End State Information

End State Names and Descriptions

File Name:
xxxxxx.ESD

File Format:
family =
name,description
... ,...

where:

family	- 16 character	Family name
name	- 16 character	End state name
description	- 60 character	End state description

End State Information

File Name:
xxxxxx.ESI

File Format:
family =
***** will be defined later *****

End State Textual Information

File Name:
xxxxxx.EST

File Format:
family, end state =
-- text --

where:

family	- 16 character	Family name
end state	- 16 character	End State name

General Rules:

1. File is standard ASCII text.
Single spaced.
Start of paragraph indented 5 spaces.
Blank line between paragraphs.
Upper and lower case.

Sequence Information

Sequence Names and Descriptions

File Name:
xxxxxx.SQD

File Format:
family,eventtree =
name,description
... ,...

where:

family	- 16 character	Family name
eventtree	- 16 character	Event tree name
name	- 16 character	Sequence name
description	- 60 character	Sequence description.

Sequence Cut Sets

File Name:
xxxxxx.SQC

File Format:
family, event tree, sequence =
eventname * eventname +
eventname * eventname * eventname *
eventname +
eventname * eventname.

where:

family	- 16 character	Family name
event tree	- 16 character	Event tree name
sequence	- 16 character	Sequence name
eventname	- 16 character	Event names in the cutset.

General Rules:

1. Events in a cut set are separated by asterisks ("*"). Spaces are ignored.
2. Cut sets are separated by a plus ("+").
3. A period (".") denotes the end of the sequence.
4. Complimented events are preceded by a slash ("/").
5. Event names are a maximum of 16 characters including the "/" character for complimented events.
6. A line starting with an asterisk ("*") is a comment.

Sequence Attributes

File Name:
xxxxxx.SQA

File Format:
family, event tree =
name,endstate,mincut,mission,procut,sample,seed,size,cuts,
events,value1,...,value9
... , ... , ... , ... , ... , ... , ...

where:

family	- 16 character	Family name
event tree	- 16 character	Event tree name
name	- 16 character	Sequence name
endstate	- 16 character	End State name
mincut	- Floating point	mincut upper bound
mission	- Floating point	Mission time in hours
procut	- Floating point	Probability cut off value
sample	- Integer 4	Sample size
seed	- Integer 8	Random number seed
size	- Integer 2	Size cut off value
cuts	- Integer 5	Base number of cutsets
events	- Integer 5	Base number of events
value	- Floating point	Base uncertainty values

General Rules:
1. If a field is unknown leave the comma as a placeholder.

Sequence Textual Information

File Name:
xxxxxx.SQT

File Format:
family, event tree, sequence=
--- text ---

where:

family	- 16 character	Family name
sequence	- 16 character	Sequence name
event tree	- 16 character	Event tree name

General Rules:
1. File is standard ASCII text.
Single spaced.
Start of paragraph indented 5 spaces.
Blank line between paragraphs.
Upper and lower case.

Piping and Instrumentation Diagrams

P&ID

File Name:

xxxxxxx.PID

File Format:

(P&ID Editor format).

MAR - D

File Extension Descriptions

Family

*.FAD	Family Name and Description
*.FAA	Family Attributes
*.FAT	Family Textual Information

Basic Event

*.BED	Basic Event Names and Descriptions
*.BEI	Basic Event Failure Rate Information
*.BEA	Basic Event Attribute Codes

Event Attributes

*.FMD	Failure Mode Descriptions
*.CTD	Component Type Descriptions
*.STD	System Type Descriptions
*.LCD	Location Descriptions
*.CAD	Class Attribute Descriptions

Fault Tree

*.FTD	Fault Tree Names and Descriptions
*.DLS	Fault Tree Graphics
*.FTL	Fault Tree Logic
*.FTC	Fault Tree Cut Sets
*.FTA	Fault Tree Attributes
*.FTT	Fault Tree Textual Information

Event Tree

*.ETD	Event Tree Names and Descriptions
*.ETA	Event Tree Attributes
*.ETG	Event Tree Graphics
*.ETL	Event Tree Logic
*.ETT	Event Tree Textual Information

End State

*.ESD	End State Names and Descriptions
*.ESI	End State Information
*.EST	End State Textual Information

Sequence

*.SQD	Sequence Names and Descriptions
*.SQC	Sequence Cut Sets
*.SQA	Sequence Attributes
*.SQT	Sequence Textual Information

P & ID

*.PID	Piping and Instrumentation Diagrams
-------	-------------------------------------