Probabilistic Risk Assessment (PRA): The basis for recognizing emerging operational risks

Prepared by
Roger L. Boyer, CRE
Chief, Analysis Branch

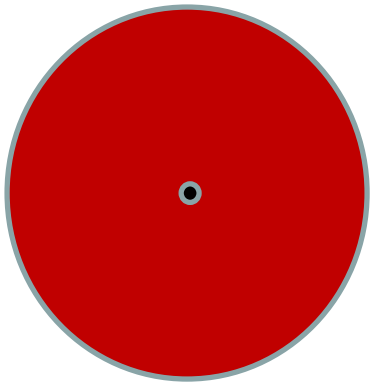
NASA Johnson Space Center
Safety & Mission Assurance (S&MA)

Prepared for
Operational Excellence & Risk Management

February 7, 2017

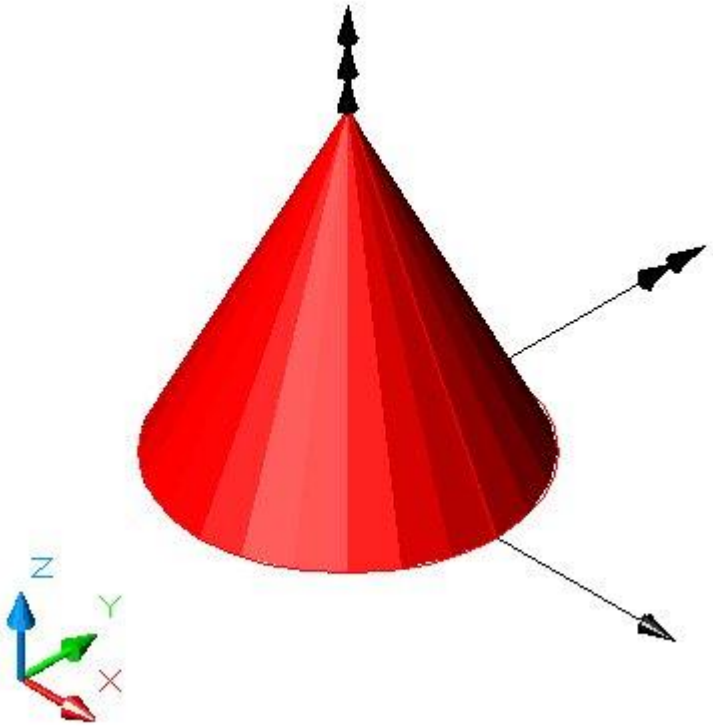
Using different views in analysis

What does this look like?



A circle with a dot in the center?
A sphere with a hole through the center?

It could be this…



Or it could be this…

A single view can mislead you…

Conclusion

As designers, you have an arsenal of tools, techniques, and personnel available to you.

Given your available budget and time, we must be smart and efficient in how and what we do. That's where you can make a difference.

Questions?

Why are we here?

WORKSHOP G: Probabilistic risk assessment: The basis for recognizing emerging operational risks

- During this session we will discuss how a systematic and comprehensive methodology to evaluate risks associated with complex engineering and technological systems can help companies identify emerging risk to their critical operations.

- Through the use of examples, we will explore how specific tools and processes can help approach operational risks with:
  - A quantitative evaluation of system safety
  - Identification, selection, and screening of initiating events
  - Definition and modeling scenarios, Initiating and Pivotal Events, Modeling & Data development, and risk quantification & uncertainty analysis
  - Risk importance ranking and cutset analysis for risk reduction and communication

Why are we here?

- PRA is one of the tools in our S&MA toolbox. It provides both depth and width in evaluating systems, vehicles, vessels, facilities, and missions.
- It's been used successfully in several industries, such as commercial nuclear power, aerospace, transportation, chemical, and medical.
- NASA continues to get budgets with high expectations from the public. S&MA must continue to do its job with less, thus we have to be smarter and more efficient.
- Today's workshop is to help take you to the next level in understanding this tool and how to use it.

  - ✓ When to do a PRA?
  - ✓ How to support/perform it?
  - ✓ How to recognize a good one?
  - ✓ How to use it in your risk-informed decision making process?

The PRA Team

The PRA Team

- **A PRA system analysis team includes both system domain experts and PRA analysts. The key to success is multi-way communication between the PRA analysts, domain experts, and management.**
- **A majority of PRA analysts have engineering degrees with operations and/or design backgrounds in order to understand how systems work and fail. This is essential in developing the failure logic of the vehicle or facility.**
- **Good data analysts understand how to take the available data to generate probabilities and their associated uncertainty for the basic events that the modelers can use or need.**
- **Building or developing a PRA involves:**
  - understanding its purpose and the appropriate modeling techniques,
  - designing how it will serve that purpose,
  - populating it with the desired failure logic and probabilities, and
  - trouble shooting it (nothing works the first time)

PRA Overview

Questions a PRA can answer for your organization:

- ✓ “What could go wrong and what are the consequences?”
- ✓ “What is the likelihood of an undesirable event?”
- ✓ “Where should I focus resources to reduce overall risk?”
- ✓ “What are the uncertainties of my processes and systems?”

Applicable Industries

PRA Overview

NEW DEVELOPMENTS

The ideal time to conduct a PRA is at the beginning of the design process to incorporate the necessary safety and risk avoidance measures throughout the development phase at minimal cost.

EXISTING SYSTEMS

PRA can be applied to existing systems to identify and prioritize risks associated with operations. Risk assessments can evaluate the impact of system changes and help avoid compromises in quality or reliability while increasing productivity.

INCIDENT RESPONSE

In the event of unexpected downtime or an accident, our team can assess the cause of the failure and develop appropriate mitigation plans to minimize the probability of comparable events in the future.

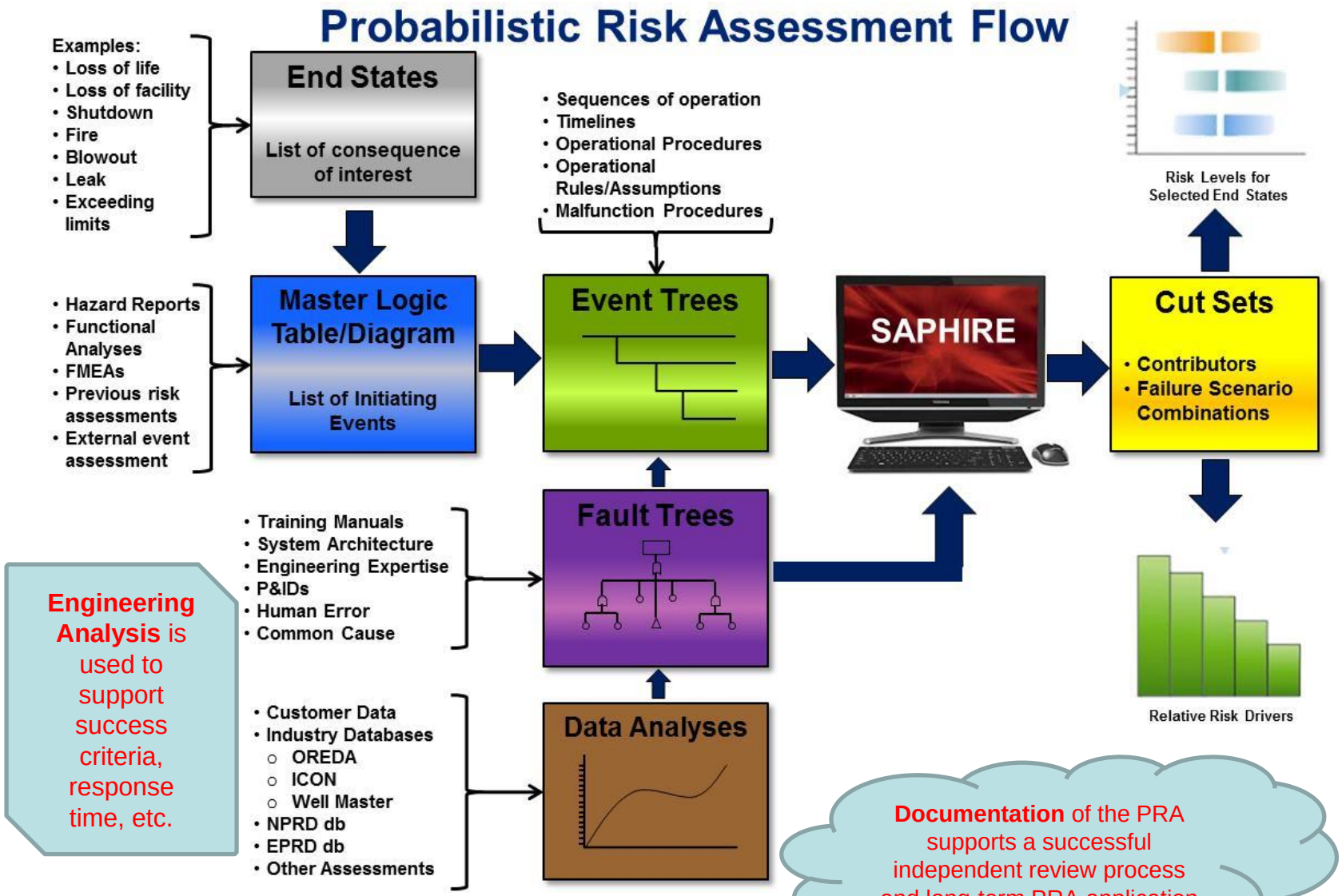
In a nutshell, PRA can be applied from concept to decommissioning during the life cycle, including design and operations.

PRA Overview

What is PRA?

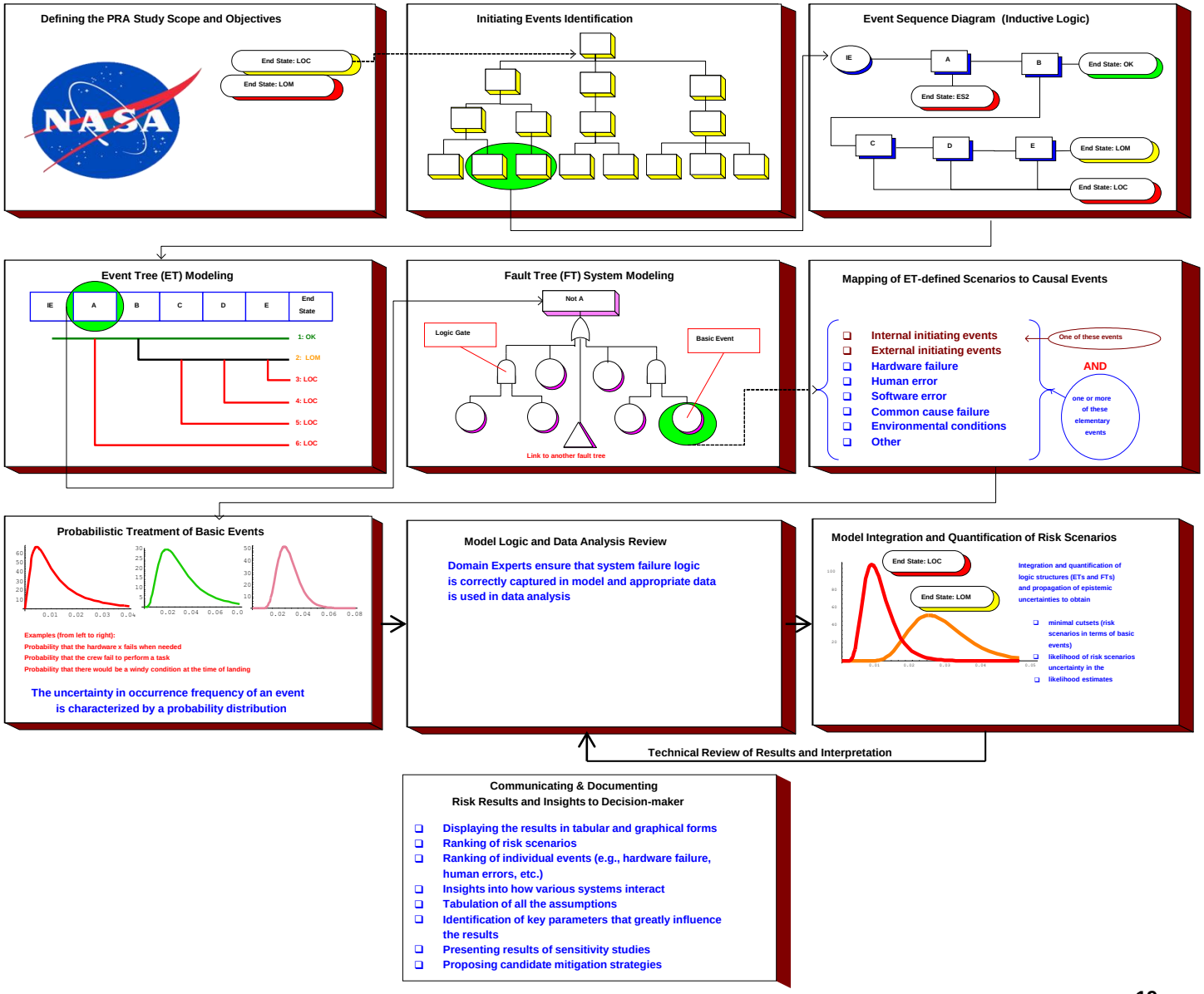
- **PRA is a comprehensive, structured, and disciplined approach to identifying and analyzing risk in engineered systems and/or processes. It attempts to quantify rare event probabilities of failures. It attempts to take into account all possible events or influences that could reasonably affect the system or process being studied. It is inherently and philosophically a Bayesian methodology. In general, PRA is a process that seeks answers to three basic questions:**
  - ✓ **What kinds of events or scenarios can occur (i.e., what can go wrong)?**
  - ✓ **What are the likelihoods and associated uncertainties of the events or scenarios?**
  - ✓ **What consequences could result from these events or scenarios (e.g., Loss of Crew and Loss of Mission)?**
- **There are other definitions**
- **The models are developed in "failure space". This is usually different from how designers think (e.g. success space).**
- **PRAs are often characterized by (but not limited to) event tree models, fault tree models, and simulation models**

PRA Process


Probabilistic Risk Assessment Flow
Examples:
• Loss of life
• Loss of facility
• Shutdown
• Fire
• Blowout
• Leak
• Exceeding limits
End States
List of consequence of interest
• Sequences of operation
• Timelines
• Operational Procedures
• Operational Rules/Assumptions
• Malfunction Procedures
Risk Levels for Selected End States
• Hazard Reports
• Functional Analyses
• FMEAs
• Previous risk assessments
• External event assessment
Master Logic Table/Diagram
List of Initiating Events
Event Trees
SAPHIRE
Cut Sets
• Contributors
• Failure Scenario Combinations
• Training Manuals
• System Architecture
• Engineering Expertise
• P&IDs
• Human Error
• Common Cause
Fault Trees
Relative Risk Drivers
• Customer Data
• Industry Databases
○ OREDA
○ ICON
○ Well Master
• NPRD db
• EPRD db
• Other Assessments
Data Analyses
Engineering Analysis is used to support success criteria, response time, etc.
Documentation of the PRA supports a successful independent review process and long-term PRA application

PRA Development Process

PRA Development Process



PRA Development Process (2)

- **Defined the scope of the PRA**
  - Start with the end in mind or the question you want answered. For example, loss of hydrocarbon containment and loss of life failure end states
  - Define mission scope,
  - Establish the mission/operational phases and layout the mission level event trees and corresponding top events to be analyzed

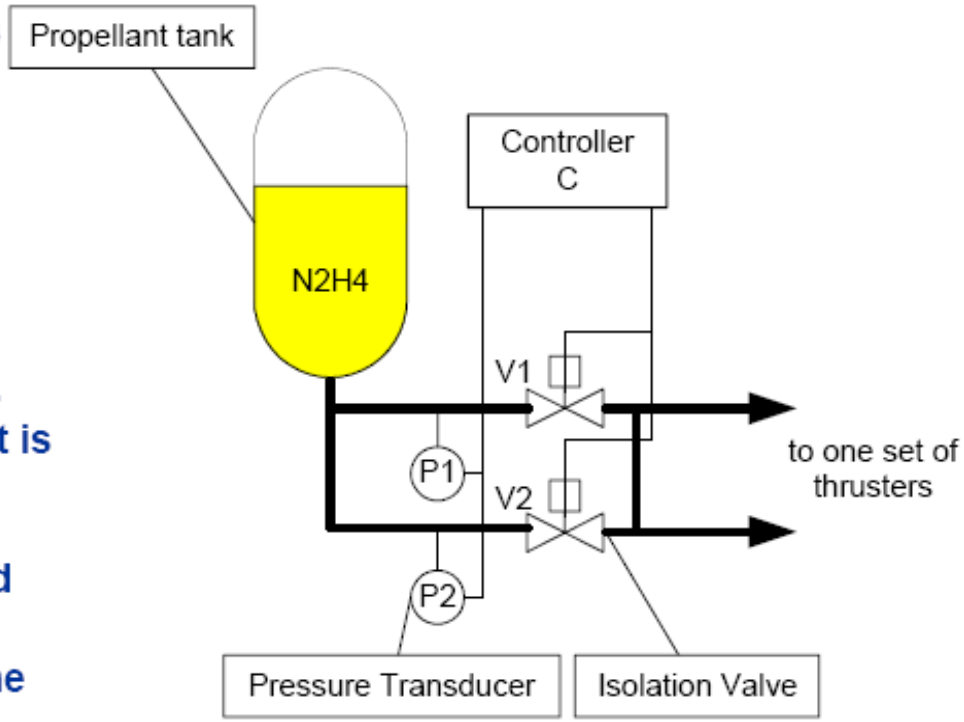
- **Develop logic models**
  - Assign top events to system analysts for each subsystem and work with domain experts to develop fault trees
  - System analysts work with data analysts and domain experts to determine level of detail and failure logic (develop fault trees to the level that data exists)
  - Obtain appropriate project office concurrence of system models (fault trees)

PRA Development Process (3)

- **Develop failure data into failure probabilities**
  - Obtain specific failure history or best available generic data
  - Data analysts calculate failure probabilities based on best available data and approved methods
- **Quantify the model, perform sanity checks, re-iterate until Team is in agreement**
  - Quantify the integrated model and perform sanity checks to determine which simplifying model assumptions need to be re-evaluated, where uncertainties need to be narrowed, where additional deterministic analyses are needed
- **Shares results with program and projects**
  - Risk ranking and risk insights
  - Incorporate feedback into PRA and into program/project design/ops
  - Maintain “Living PRA” to represent new program information (data updates) and evolving model scope

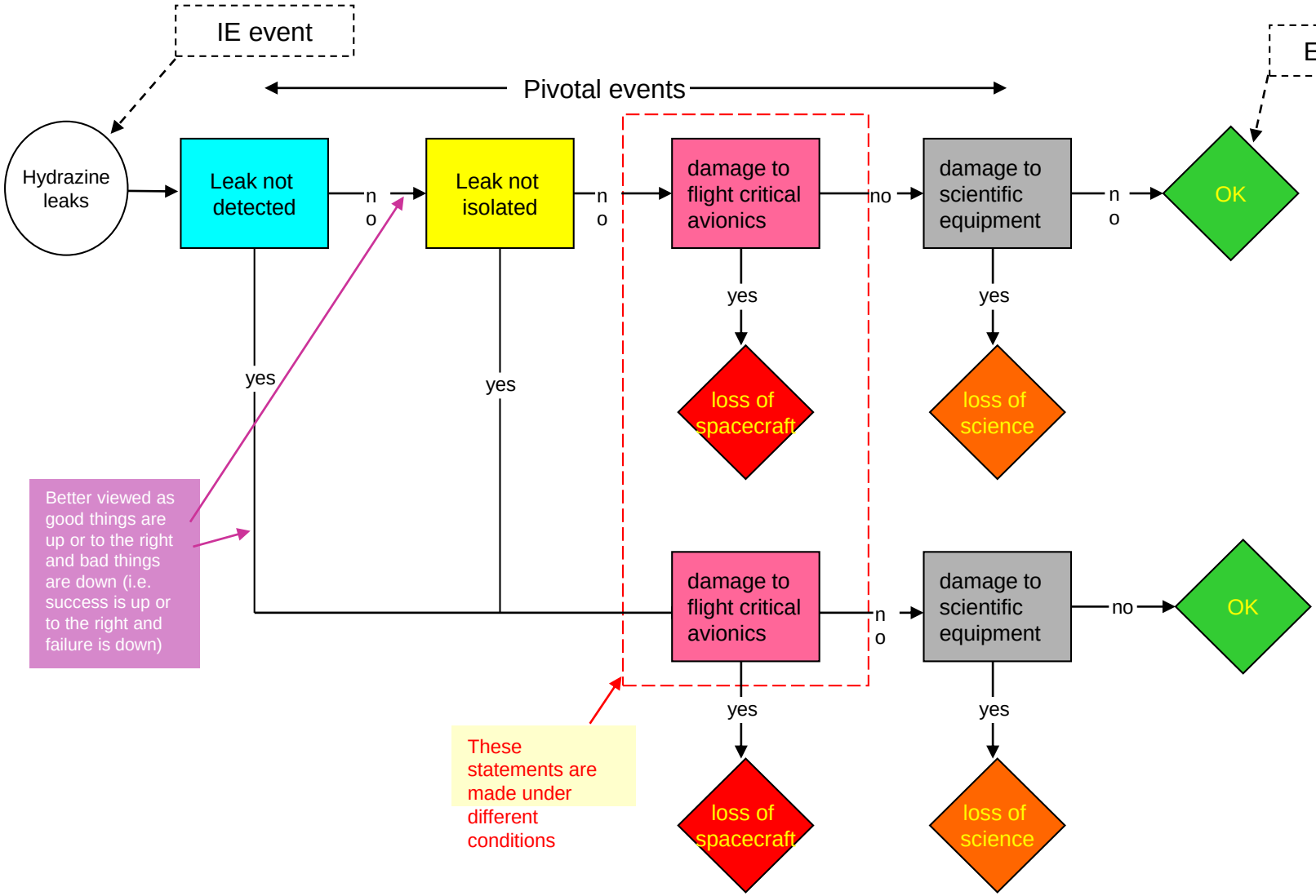
Simple Example of a Small PRA model

- The spacecraft is designed with two redundant sets of thrusters (independent of each other)
- Each propellant distribution module consists a hydrazine tank, filters, distribution lines, normally-open isolation valves, sensors, heaters, etc. (only components that affect mitigation of leaks are shown)
- When thruster operation is needed, the controller opens the solenoid valves (not shown) to allow hydrazine to flow
- The controller monitors the pressure of feed-lines via pressure transducers (P1 and P2). It is designed to differentiate between the normal thruster operation and a leak
- In the event of a leak, isolation valves (V1 and V2) should both close
- Successful termination of the leak leads to the loss of one but not both, thruster sets
- Failure to terminate the leak can cause damage to the flight critical avionics and/or damage to scientific equipment:
  - Hydrazine acts as a wire stripper and is corrosive

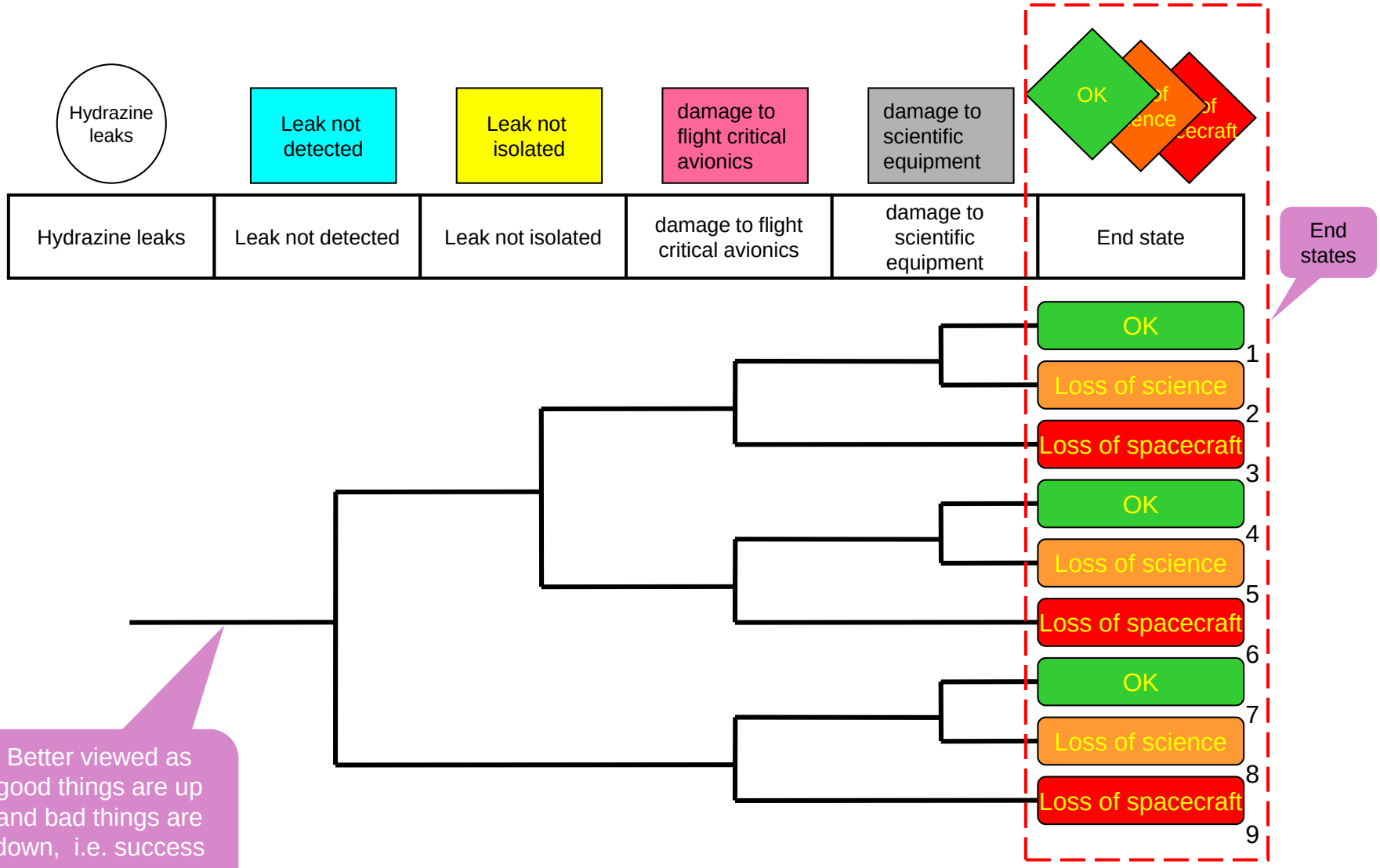


Simplified Schematic of Propellant Distribution Module

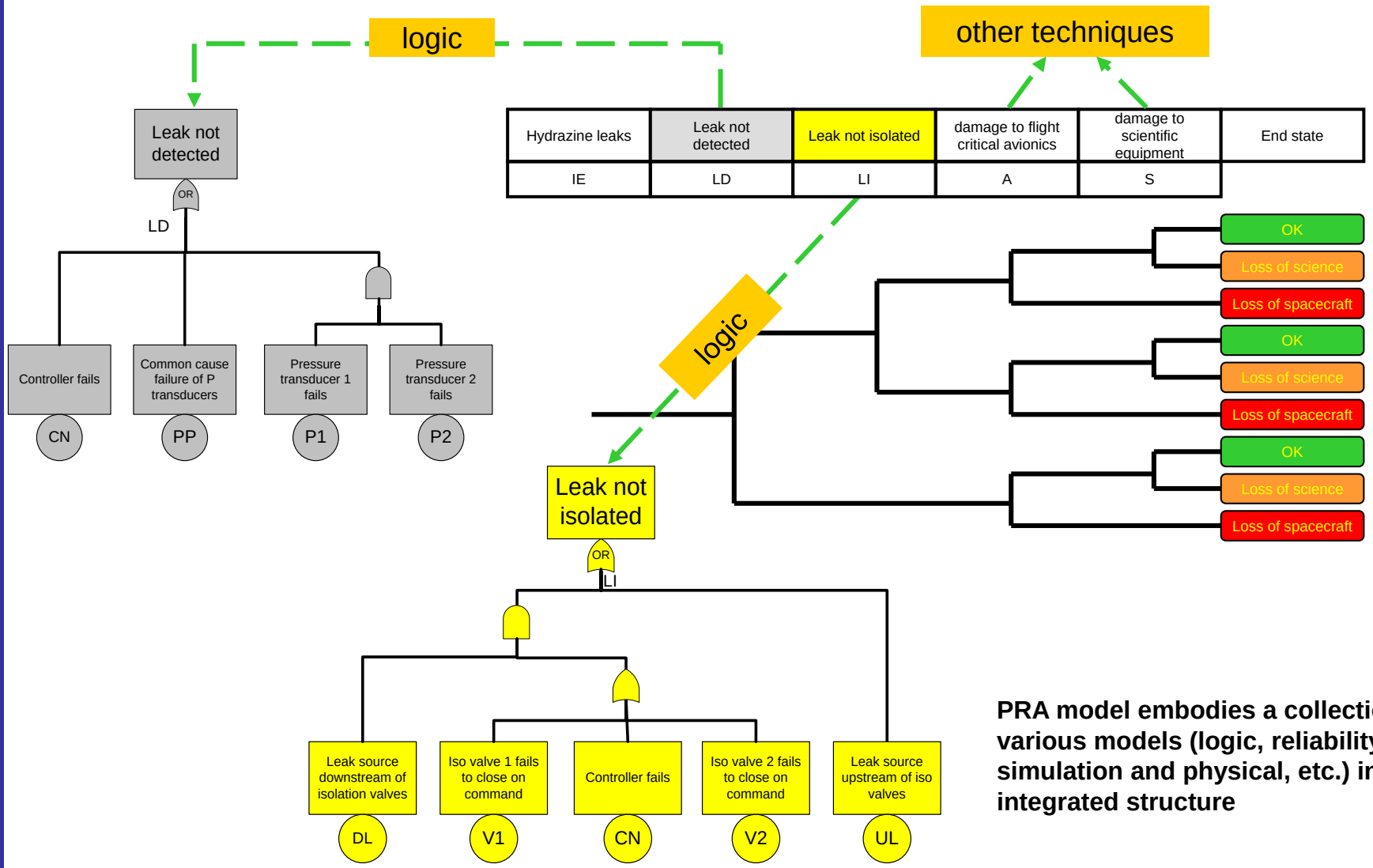
Example of Event Sequence Diagram (ESD)



The ESD Translated Into an Event Tree



Fault Trees Are Attached to the Event Tree



PRA model embodies a collection of various models (logic, reliability, simulation and physical, etc.) in an integrated structure

Common Cause

Common Cause

- Definition Of Common Cause Failure (CCF)
- Some basics
- Types Of CCF Models
- Examples of common cause
- Deriving common cause parameter values from data
- Examples of Beta's calculated from real data (NASA and Nuclear)
- Conclusions

Common Cause Modeling

(More details and examples on this later)

- **All large PRAs of complex and redundant machines must include "common cause" effects to be complete and accurate**
- **Common Cause are those conditions that defeat the benefits of redundancy**
  - Not "single point failures"
  - Similar to "generic cause"
- **There are three recognized ways to perform common cause modeling:**
  - The Beta Model
  - The Multiple Greek Letter Model
  - The Alpha Model
- **We use an iterative approach to modeling common cause first the Beta Model approach is used and if it shows up as a risk driver a Multiple Greek Letter Model is used**
- **Generic data from NUREG/CR-5485 for the majority of the events since there are few cases where there is enough Shuttle data to develop Shuttle specific values**
  - RCS Thrusters and ECO sensors are examples of cases where Shuttle specific data is used to calculate the common cause parameters

Common Cause Modeling (2)

HOW THE BETA MODEL APPROACH WORKS

- **Susceptibility groups (groupings of similar or identical equipment) of redundant trains or components are identified**
- **A common cause basic event is defined for these groups**
- **The common cause basic event failure rate is generated by taking the independent failure rate times a "Beta" factor.**
  - For the beta model it does not matter how many components are in the group
  - The "Beta" factor represents the probability of 2 or more failures given a failure has occurred
    - > For this reason, the Beta Model may be conservative for component groups larger than 2.
- **The "Beta" factor is taken from NUREG/CR-5485 and has a different value for "Operating" failures vs. "Demand" failures**
  - Operating failures the "Beta" value is 0.0235
  - Demand failures the "Beta" value is 0.047

Common Cause Modeling (3)

HOW THE MULTIPLE GREEK MODEL APPROACH WORKS

- Similar to the Beta Model except that the Multiple Greek Model takes credit for the full redundancy and therefore can be much more complicated
  - For a 3 component group, there is a “beta” factor and a “gamma” factor where the “beta factor is still the probability of 2 or more failures and the “gamma” factor is the probability of 3 or more failures given 2 or more failures.

Common Cause Definition

- **In PRA, Common Cause Failures (CCFs) are failures of two or more components, subsystems, or structures due to a single specific event which bypassed or invalidated redundancy or independence at the same time, or in a relatively short interval like within a single mission**
  - May be the result of a design error, installation error, or maintenance error, or due to some adverse common environment
  - Sometimes called a generic failure.
- **Common Cause, as used in PRA, is <u>not</u> a single failure that takes out multiple components such as a common power supply to computers or common fluid header to multiple pumps.**
  - Single point failures such as these are modeled explicitly in a PRA

Some Basics on PRA and Common Cause Failures

- **PRA**
  - PRA is used to perform "rare event" analysis
    - If we had 1000 Space Stations operating for 50 years each and we had lost 60 of them we would not need to do a PRA to determine what the loss of station failure rate was
    - However, we have only had one Station operating for ~ 10 years with no loss of station so methods like PRA are needed to estimate this value
  - Most of the components used in space vehicles are designed to be low failure rates and limited numbers of these components mean that an actual failure rate number is difficult to calculate from operational data (uncertainty is high!)
- **Common Cause Parameters**
  - Beta is modeled as a fraction of the total failure rate.
    - Total failure rate = Independent failure rate + common cause failure rate
    - Beta = common cause failure rate / Total failure rate
    - This is ~ to common cause failure rate / independent failure rate (when Beta is small)
  - **<u>If you have a low failure rate for a component, the common cause failure rate will be low too but could still have a high Beta factor</u>**
  - A failure rate is a rate such as Failures per hour and a Failure probability is derived by the equation of $1-e^{-lt}$ where l is the failure rate. When lt is a small value the equation can be simplified using the rare event approximation and we get Failure probability ~ lt.

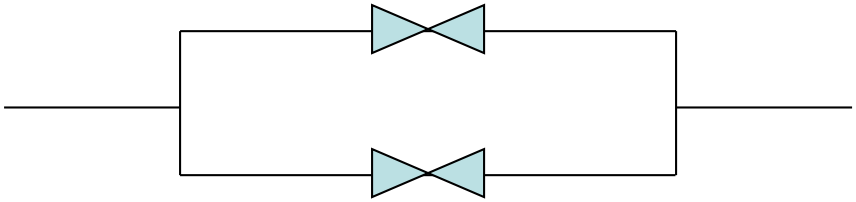
Note: **Beta is a parameter of a single modeling method, and there are several modeling methods and variations most work in similar fashion**

Types Of Common Cause Models

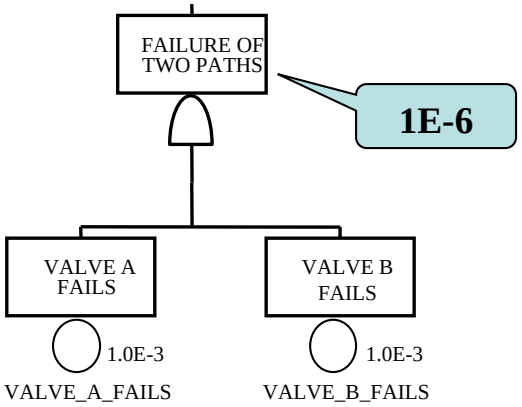
- Common Cause is modeled as a **conditional** probability, i.e. Given that a component has failed, what is the probability that another like component will fail
- Common models used are:
  - Beta (β) model – For a system with multiple like components, Beta factor is used to estimate the probability of failure of **all** components (i.e. two or more)
    - Values for Beta can range from 1 to 0.0001 (or less), but more typical values are usually between 0.1 and 0.001
  - Multiple Greek Letter (MGL) model – For systems with 3 or more like components, provides for a more explicit breakdown of possibilities, probabilities of two, three, four, etc. component failures
  - Alpha (α) model – Similar to the MGL model

Example Of Impact Of Modeling Common Cause

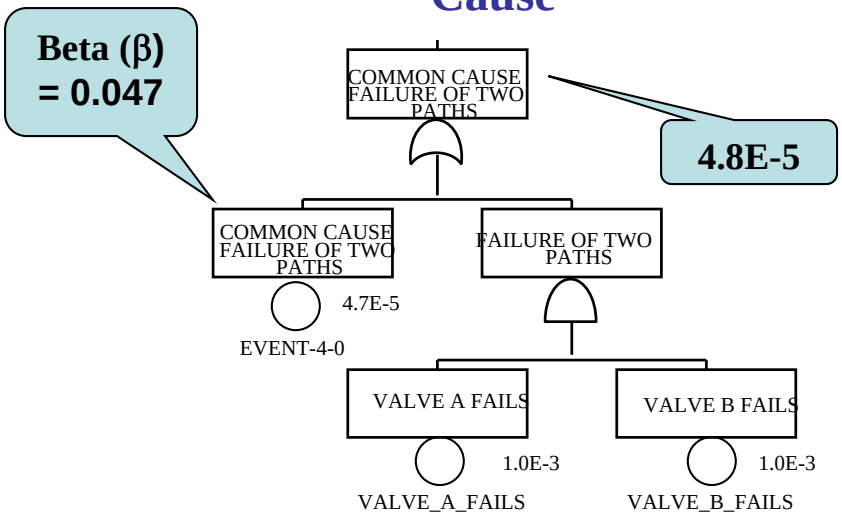
A system consisting of two trains:



Without Considering Common Cause



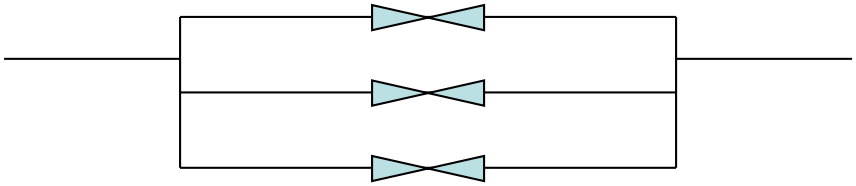
Considering Common Cause



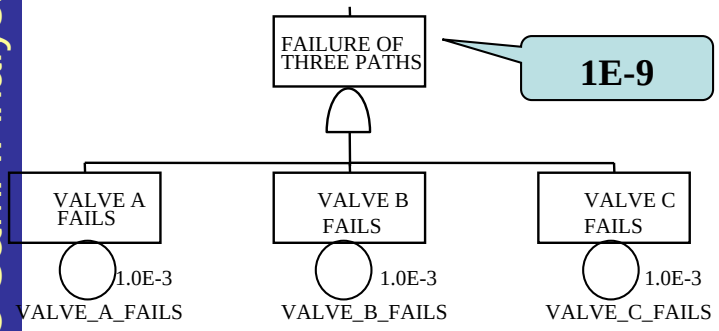
Results in a ~ 4.7E-05 Underestimate of Risk Which is 48 Times the Risk Without Considering Common Cause

Example Of Impact Of Modeling Common Cause

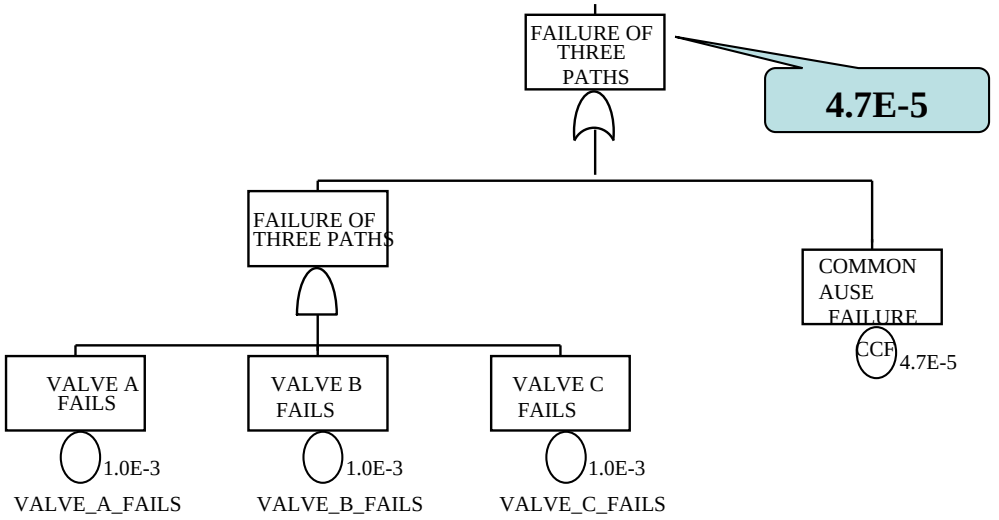
A system consisting of three trains:



Without Considering Common Cause



Considering Common Cause (Beta Model)



Results in a ~ 4.7E-05 Underestimate of Risk Which is 47,000 Times the Risk Without Considering Common Cause

Note: Using a MGL Model Would Reduce Result to 2.6E-05

Types Of Data That Exist In The Models

- **Functional – A functional failure event is generally defined as failure of a component type, such as a valve or pump, to perform its intended function. Functional failures are specified by a component type (e.g., motor pump) and by a failure mode for the component type (e.g., fails to start). Functional failures are generally defined at the major component level such as Line Replaceable Unit (LRU) or Shop Replaceable Unit (SRU). Functional failures typically fall into two categories, time-based and demand-based. Bayesian update as Shuttle specific data becomes available.**

- **Phenomenological – Phenomenological events include non-functional events that are not solely based on equipment performance but on complex interactions between systems and their environment or other external factors or events. Phenomenological events can cover a broad range of failure scenarios, including leaks of flammable/explosive fluids, engine burn through, overpressurization, ascent debris, structural failure, and other similar situations.**

- **Human – Three types of human errors are generally included in fault trees: pre-initiating event, initiating event (or human-induced initiators), and post-initiating event interactions.**

- **Common Cause – Common Cause Failures (CCFs) are multiple failures of similar components within a system that occur within a specified period of time due to a shared cause.**

- **Conditional – A probability that is conditional upon another event, i.e. given that an event has already happened what is the probability that successive events will fail**

Notional PRA Examples

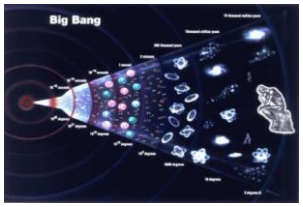
First the Math

1.0E-02 = 0.01 ➔ 1:100 (Probable) ➔ ~Shuttle Mission Risk

1.0E-06 = 0.000001 ➔ 1:1,000,000 (Improbable) ➔ having 20 coins simulaneously landing on tails

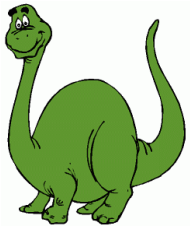
1.0E-12 = 0.000000000001 ➔ 1:1,000,000,000,000 (ridiculous)

Time Perspective



1.2×10^{14} hours ago
~14 billion years ago

4×10^{13} hours ago
~4.5 billion years ago



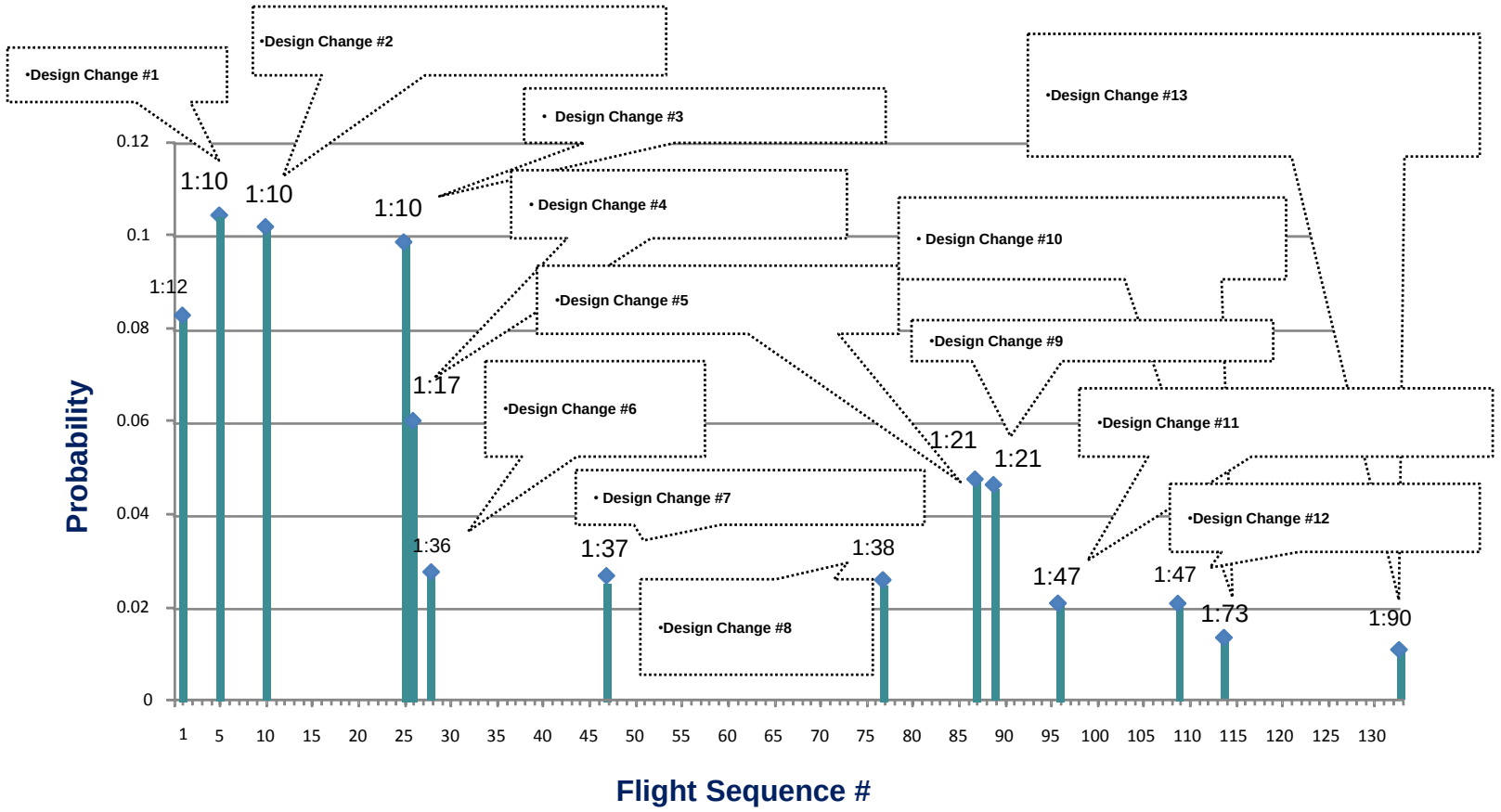
$2 \times 10^{12} – 7 \times 10^{11}$ hours ago
~228 – 80 million years ago

4×10^{8} hours ago
~46,000 years ago

2.1×10^{6} hours ago
~240 years ago

6.3×10^{5} hours ago
~72 years ago

Risk Regression Example



This chart shows how calculated risk changed following design and ops changes over a 30 year program by peeling back the "onion" (starting at the end and undoing changes). Note that risk doesn't decrease according to a nice exponential curve, but only after something fails and it gets "fixed".

Uncertainty Distribution

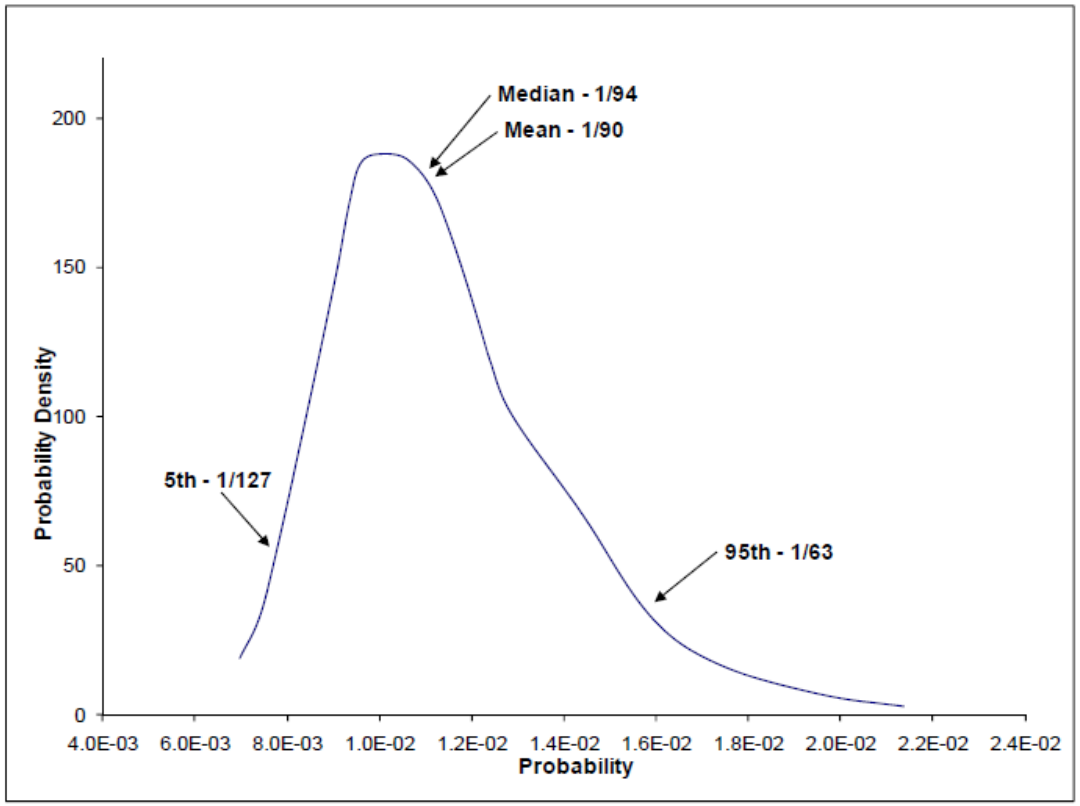
- This distribution is a representation of the uncertainty associated with a PRA's results
- The median is also referred to as the 50th percentile

Mean – 1.1E-02 (1:90)

Median – 1.1E-02 (1:94)

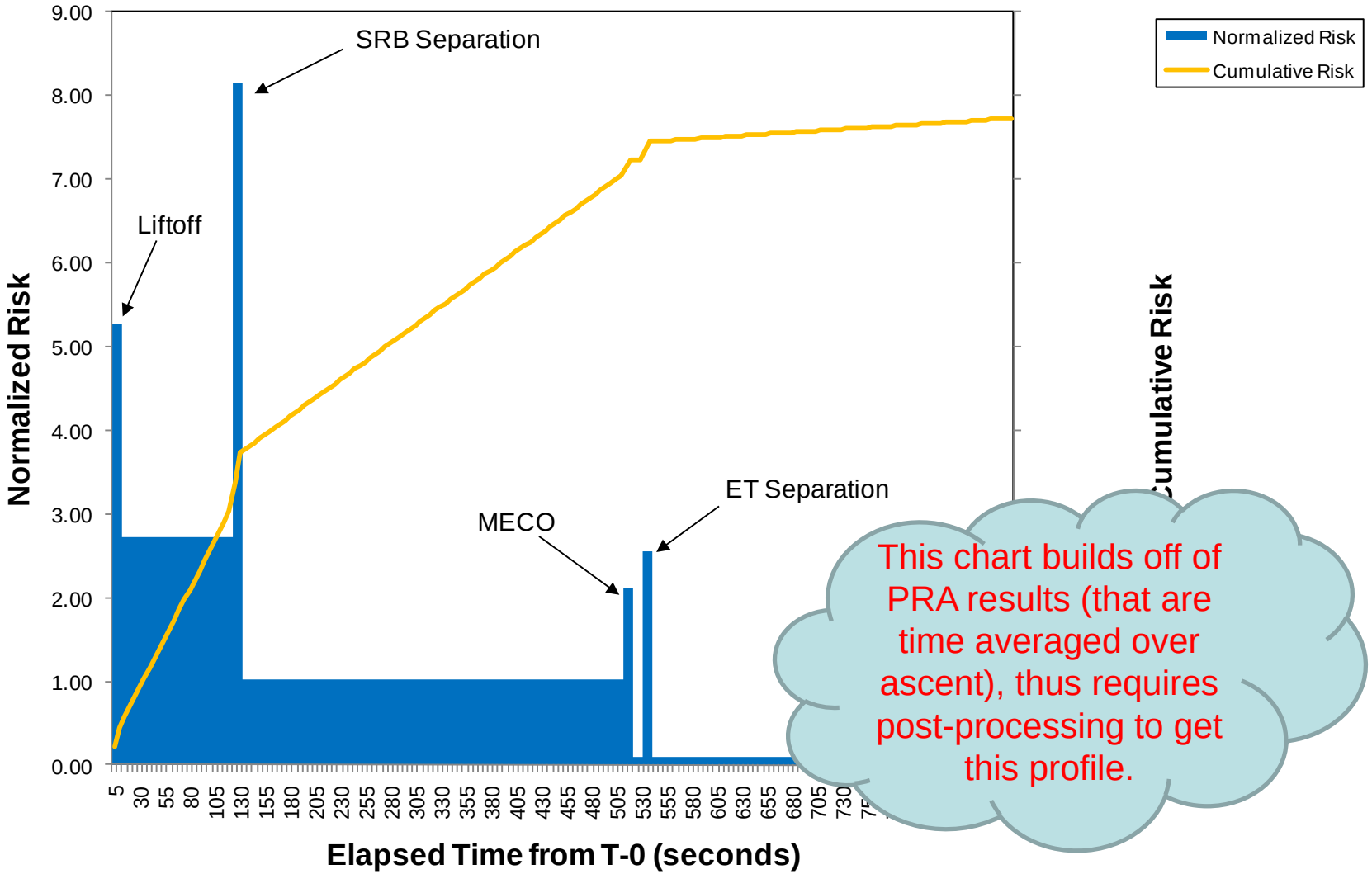
5th percentile – 7.9E-03 (1:127)

95th percentile – 1.6E-02 (1:63)



- The 5th and 95th percentile are common points on a distribution to show the range that 90% of the estimated risk lies between.
- The mean is a common measure of risk that accounts for uncertainty or this distribution, thus the value or metric used to verify LOC requirements.

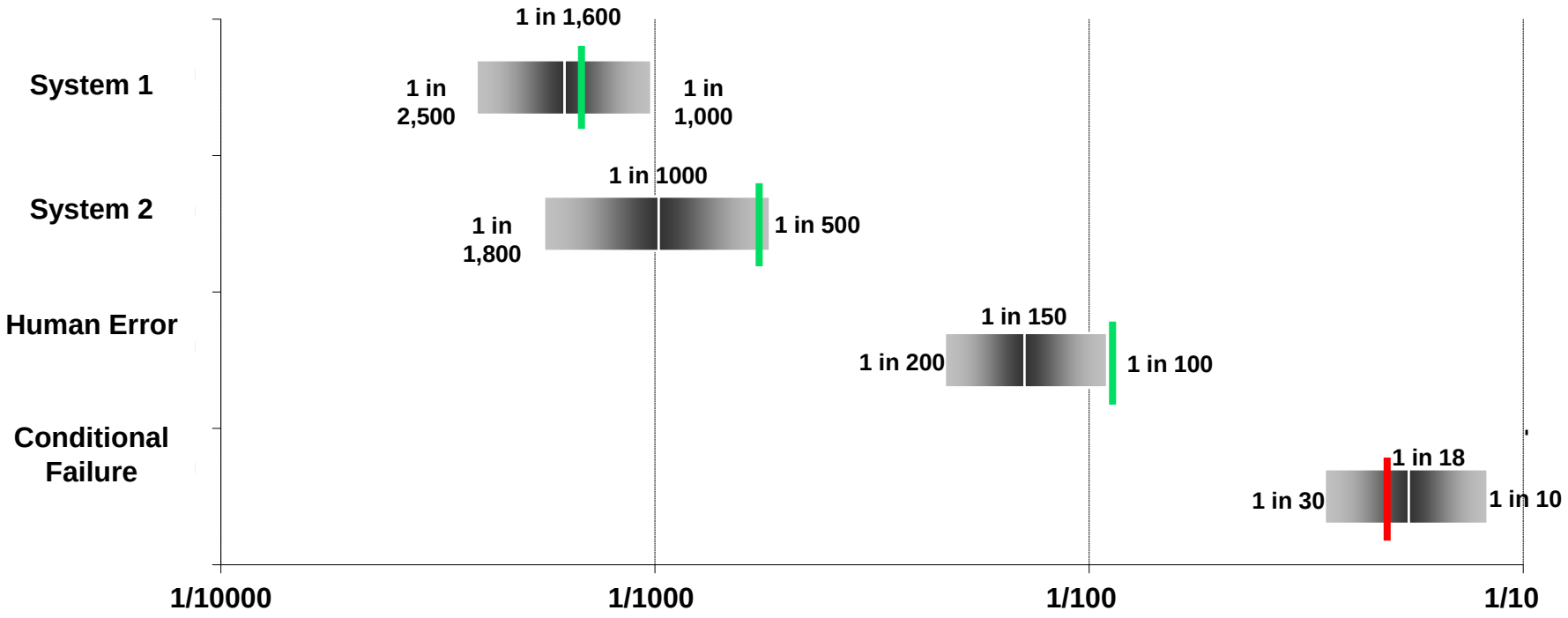
Notional Ascent Risk Profile (not a direct output of PRA)



This chart builds off of PRA results (that are time averaged over ascent), thus requires post-processing to get this profile.

Showing Uncertainty wrt Requirements

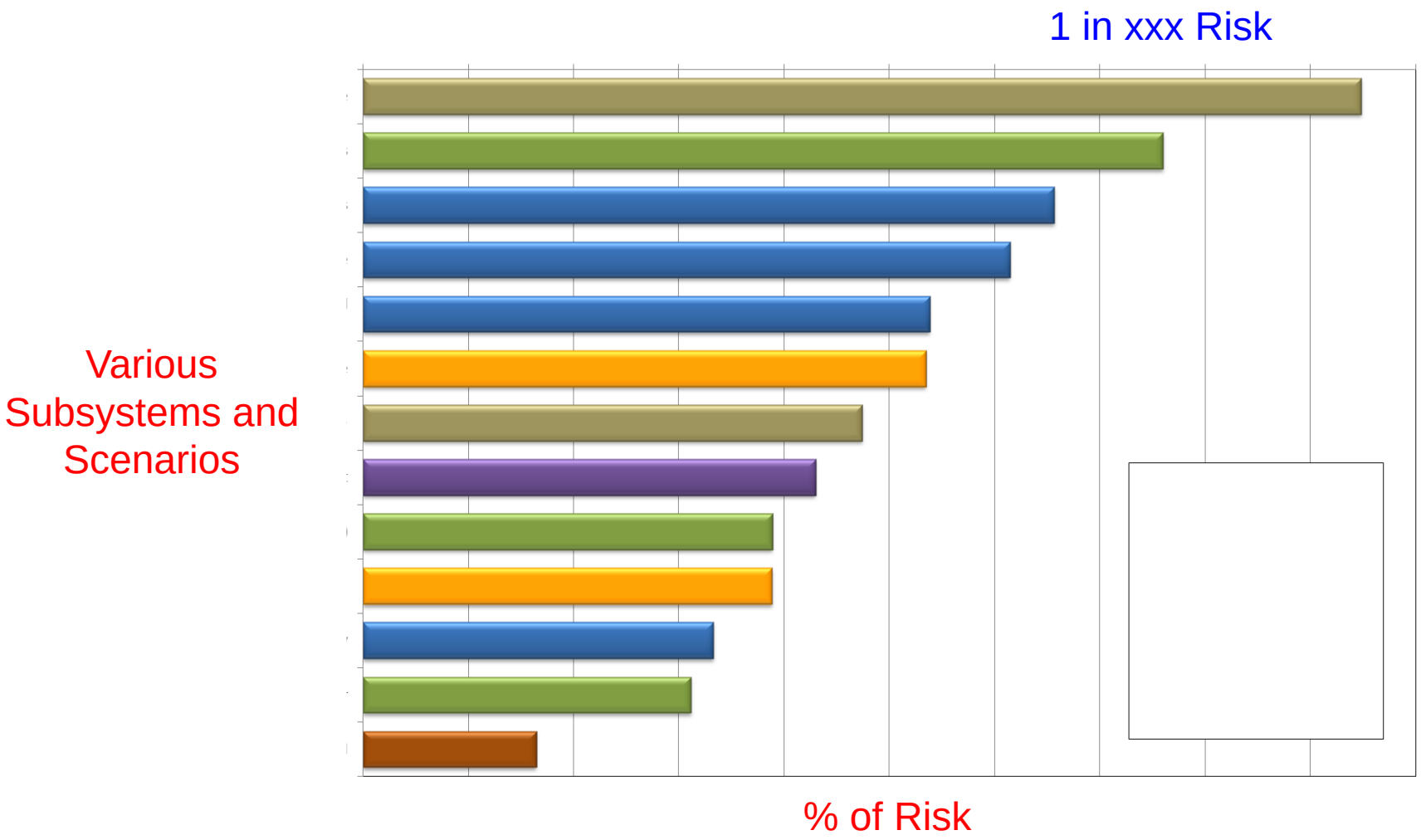
Notional



Green Bar shows Requirement Value is met
Red Bar shows Requirement Value is not met

Notional Risk Drivers via Pareto
(Top 80% of Calculated Risk)

A Pareto chart like this can be made for each project, rig, platform, etc.

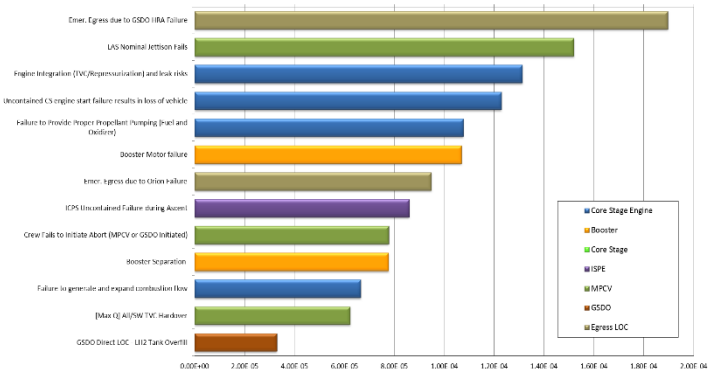


Integrated Risk-Informed Design Assessment

Risk Trade Study for Proposed Change

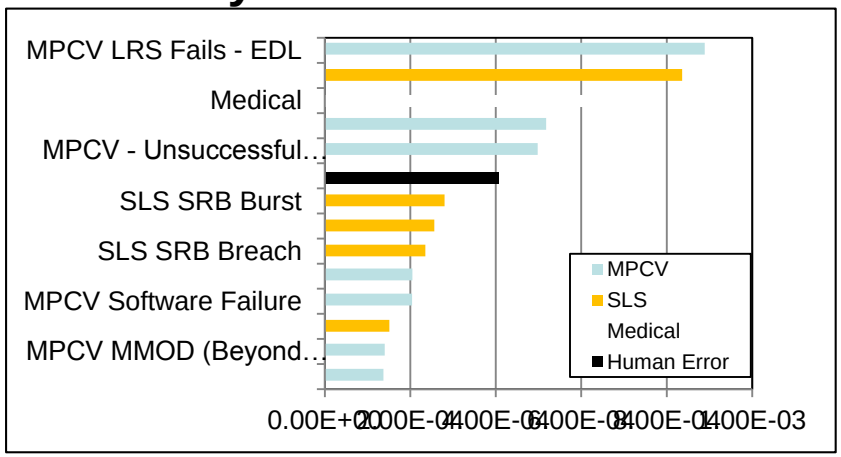
- Baseline from PRA and Achievability Study
- addresses the three scenarios where design change can reduce risk and the additional risk associated with inadvertent operation
- Result is an increase in probability of LOM due to inadvertent ops with no LOC advantage

System Risk Drivers



Baseline scenarios are #54 in Risk < 1%

Facility Risk Drivers



Baseline scenarios are #234 in Risk < 0.2%

• Bottom Line

- 20% of mission risk is due to ABC (#1 Risk) and CBA (#2 Risk)
- Baseline scenarios to be addressed by proposed change is insignificant
- Recommend spending resources on top risk drivers

When Should You Do a PRA?

- As early in the design process as you can in order to affect the design and corresponding risk with minimal cost impact (i.e. to support Risk Informed Design (RID))
- When the risk of losing the project is greater than the company can live with either due to loss of life or for environmental or economic reasons
- To support Risk Informed Decision Making (RIDM) throughout a project's life cycle from "formulation to implementation" or "concept to decommissioning"

How much does a PRA cost?

- **As you can also ask, "How much will it cost to <u>not</u> do a PRA?"**
- **The cost of a PRA is a function of the level of detail desired as well as the size/complexity of the item being assessed and the mission life cycle**
  - You should only model to the level of detail that you have data and no further. You may identify that significant risk exists at a sublevel, then your PRA is telling you that you need to study that level further. It may not be a PRA, but a reliability assessment at that time.
  - Modeling a drilling rig is on a different scale than just the BOP. However, understanding the need for a BOP can be important in its design and operation.

Absolute vs Relative Risk?

- **You may have heard, "Don't believe the absolute risk estimate, just the relative ranking".**
- **Each event in a PRA is assessed to having a probability of failure (since the PRA is performed in "failure space").**
  - these failures are combined via the failure logic which is used to determine how they are combined and the resulting scenarios.
  - the failure probabilities of each event are used to establish the probability of each scenario thus ranks the scenarios as well as being added to produce the overall risk.
  - If different approaches and methods are used (which sometimes are needed in full scope PRAs), then the absolutes can be challenged and so may their rankings. This is where experienced PRA analysts earn their pay to help minimize the difference.
- **As a result, some decision makers or risk takers want to know the overall risk, while others want to know how to reduce it by working on the top risk drivers first.**

Unknown and Underappreciated Risks

- **Risk model completeness** has long been recognized as a challenge for simulated methods of risk analysis such as PRA as traditionally practiced.

- These **methods are generally effective** at identifying system failures that result from combinations of component failures that propagate through the system due to the functional dependencies of the system that are represented in the risk model.

- However, they are typically ineffective at identifying system failures that result from **unknown or underappreciated (UU)** risks, frequently involving complex intra- and inter-system interactions that may have little to do with the intentionally engineered functional relationships of the system.

Unknown and Underappreciated Risks (Cont'd)

- Earlier in 2009, the NASA Advisory Council noted the following set of contributory factors:
  - Inadequate definitions prior to agency budget decision and to external commitments
  - optimistic cost estimates/estimating errors
  - inability to execute initial schedule baseline
  - Inadequate risk assessments
  - higher technical complexity of projects than anticipated
  - changes in scope (design/content)
  - Inadequate assessment of impacts of schedule changes on cost
  - annual funding instability
  - eroding in-housetechnicalexpertise
  - poor tracking of contractor requirements against plans
  - Reserve position adequacy
  - lack of probabilistic estimating
  - "go as you can afford" approach
  - lack of formal document for recording key technical, schedule, and programmatic assumptions.

Why Do PRA?

- **What does a PRA tell you?**
  - In a large percentage of cases, the PRA tells you, or confirms for you, what you thought you already knew
    - What it also does in these cases is document in a meaningful way **why** you thought this was true
    - PRAs systematically connect design, logic, operations, Human interaction and external influences for all aspects of large complex machines to detect dependencies and effects that the human mind just could not track and grasp on its own
  - In a small percentage of cases, the PRA results show something significant that you didn't know
    - In these cases you may have a false sense of understanding and in fact the PRA has pointed out something that has been overlooked **or:**
    - Your gut feel is correct and there is a problem with the way something is modeled in a PRA

- **What does performing the PRA tell you?**
  - PRAs are recognized as tools that have enhanced the understanding between operations and engineers as to how the equipment really works, is used, and fails by promoting communication across disciplines and organizations.
  - It also gives a framework for resolving problems and failures.

Why Do PRA? (Cont'd)

- **PRAs are used to model and quantify rare events**
  - If we had 100,000 space stations operating for 40 years each with a catastrophic failure of 500 of them, we could do pretty standard statistics to estimate the probability of catastrophic failure of a space station.
    - > However, we have only one space station and it has had minimal experience and no catastrophic failures. Therefore, there will rarely be any statistically significant data since it is in rare event territory.
    - > PRA takes into account external events
      - = Micro-meteoroid and orbital debris (MMOD)
      - = Fire, etc.
    - > PRA takes into account Human Error and Common Cause
    - > PRA links functional dependency of systems and operations
    - > PRA performs uncertainty analysis

In Closing

- **There is much more to know about PRA than what you've seen today. This presentation was to give you insight in order to ask the right questions when you are trying to decide:**
  - whether you need a PRA or not,
  - is it being performed properly and by qualified analysts,
  - is it answering the question(s) you need answered.

- **PRA (with the help of deterministic analyses) identifies and ranks the risk contributors, the FMEA analysts and Reliability Engineers can help solve the problem by focusing on the top risk drivers.**

Backup Charts

Some Background

- **In late fifties / early sixties Boeing and Bell Labs developed Fault Trees to evaluate launch systems for nuclear weapons and early approaches to human reliability analysis began**
- **NASA experimented with Fault Trees and some early attempts to do Probabilistic Risk Assessment (PRA) in sixties (most notably on the Apollo Program) but then abandoned it and reduced quantitative risk assessment**
- **Nuclear power industry picked up the technology in early seventies and created WASH-1400 (Reactor Safety Study) in mid seventies.**
  - This is considered the first modern PRA
  - Was shelved until Three Mile Island (TMI) incident happened in 1979. It was determined that the WASH-1400 study gave insights to the incident that could not be easily gained by any other means.
- **PRA is now practiced by all commercial nuclear plants in the United States and a large amount of data, methodology and documentation for PRA technology has been developed by the industry and the Nuclear Regulatory Commission (NRC)**
  - All new Nuclear Plants must license their plants based on PRA as well as “Defense In Depth” concepts.
  - The NRC practices its oversight responsibility of the commercial nuclear industry using a “Risk” based approach that is heavily dependent on PRA.

Acronyms and Definitions

1. **Cut set:** Those combinations of items that can cause a failure of the type that you are interested in. A "minimum cutset" is the minimum combination of items necessary to cause the failure of interest.
2. **End State:** The consequence of interest that is defined for what your model is supposed to calculate (sometimes will be referred to as a Top event or Figure of merit depending on model type).
3. **Top event (Top):** The top event in a fault tree or a pivotal event in an event tree. If an event tree uses a linked fault tree to calculate a pivotal event then the pivotal event name and Fault tree "Top" name need to be identical.
4. **MLD:** Master Logic Diagram. Used to identify all possible initiators.
5. **Event Tree:** A logic tool that is used to model inductive logic and quantify models using Boolean logic. Can be linked to other event trees and can use fault trees linked to it.
6. **Fault Tree:** A logic tool that is used to build deductive models of equipment or processes and is quantified with Boolean Logic. Can be linked to Event Trees for a linked fault tree model. Built from top down and quantified from bottom up.
7. **PRA:** Probabilistic Risk Assessment: A technique used for evaluating rare events for complex systems or processes. Attempts to account for all possible events that can cause the "end state", "Top event", "Figure of Merit". Uses fault trees, event trees and other methods to "infer" the probability of events of interest. **Better definition later.**
8. **Rare Event:** An event that has a small probability of happening. From a data point of view, it will have never been seen in practice or seen only rarely. It will not have enough data to be statistically significant. From the "rare event approximation point of view it is a probability that is 0.1 or less.

Acronyms and Definitions

(continued)

9. **LOC: Loss of Crew:** A common "end state", "top event" consequence, or "Figure of Merit" that we are interested in at NASA.
10. **LOM: Loss of Mission;** A common "end state", "top event", consequence, or "Figure of Merit" that we are interested in at NASA.
11. **Risk:** Probability or Frequency, times consequences
12. **"And" gate:** A logic symbol used in Fault Trees that multiplies inputs to it. In Boolean algebra it defines the "intersection" of events that are put into it.
13. **"Or" gate:** A logic symbol used in Fault trees that adds inputs to it. More accurately, in Boolean Algebra" it is the "union" of events that are put into it
14. **Bathtub Curve:** This is a curve shaped like a bathtub that represents infant mortality or break-in failures early in a component or systems life and wear-out or aging late in life with a relatively constant or flat line connecting them. The flat line or constant failure rate implies that failure rates are random and independent of time.
15. **Infant mortality:** The portion on the bathtub curve that is on the front end showing that failure rates are improving (becoming smaller) as time increases.
16. **Aging:** The Portion on the Bathtub curve that is on the back end that shows the failure rates increasing as components wear out or age.
17. **Exponential Distribution:** This is the distribution or equation that we use to represent the flat part of the bathtub curve (constant failure rate) and our PRA models that rely on the failure rates being random with respect to time. For reliability it is e^{-lt} and in failure space it is $1-e^{-\lambda t}$

Acronyms and Definitions

(continued)

18. **Time Rate of Failure:** Failures that are defined as a rate of failure per time interval (e.g. failures per hour)
19. **Demand Failure:** Failures that are defined as a failure per demand.
20. **Conditional Probability:** This is a probability of occurrence that is pre-conditioned on a specific set of circumstances that precedes it or is concurrent with it.
21. **Frequency:** This is a rate (usually per time but can defined per other parameters such as demands etc.). This is a number greater than 0 but not necessarily less than 1.
22. **Probability:** Dimensionless number between 0 and 1. Describes the likelihood of something happening.
23. **Minimal Cutset:** A "minimum cutset" is the minimum combination of items necessary to cause the failure of interest.
24. **ESD: Event Sequence Diagram:** This is a tool sometimes used to help explain the flow of an event or events and can be directly represented by an event tree. It uses inductive logic. Relatively few computer software programs will quantify ESDs.
25. **Lambda:** This is a rate of failure. Often uses the Greek symbol l. Most of the time this will be a time rate of failure but can also be used to represent a "demand rate of failure".
26. **λ:** Greek letter Lambda often used to show a failure rate.

Acronyms and Definitions

(continued)

27. **Lognormal Distribution:** This is a distribution of events that if graphed on log paper it would show a normal distribution. It is a distribution often used in the PRA world to define the uncertainty of Lambda (λ).
28. **EF (Error Factor):** This is a parameter used to help define the width of a lognormal distribution. It is defined as the 95th/50th = 50th/5th = Square root of 95th/5th . We will often times approximate a result of an uncertainty evaluation with a Lognormal distribution when it is in fact not a lognormal or any other kind of distribution but a lognormal does a good job of approximating it. In such cases we always try and use the definition of EF= Square root of 95th/5th.
29. **Fussel Vessely (FV):** Fussel Vesely importance measure. Represents how much of a components failure is contributing to the Top event or end state. Often expressed as a percentage it is not really and will be covered later.
30. **Risk Increase Ratio (RIR):** This is another importance measure that will tell you how much a Top Event or End State will increase if you set an items probability of failure to 1 and recalculate the end state or top event. It is equivalent to RAW.
31. **Risk Achievement Ration (RAW):** This is another importance measure that will tell you how much a Top Event or End State will increase if you set an items probability of failure to 1 and recalculate the end state or top event. It is equivalent to RIR.

Acronyms and Definitions

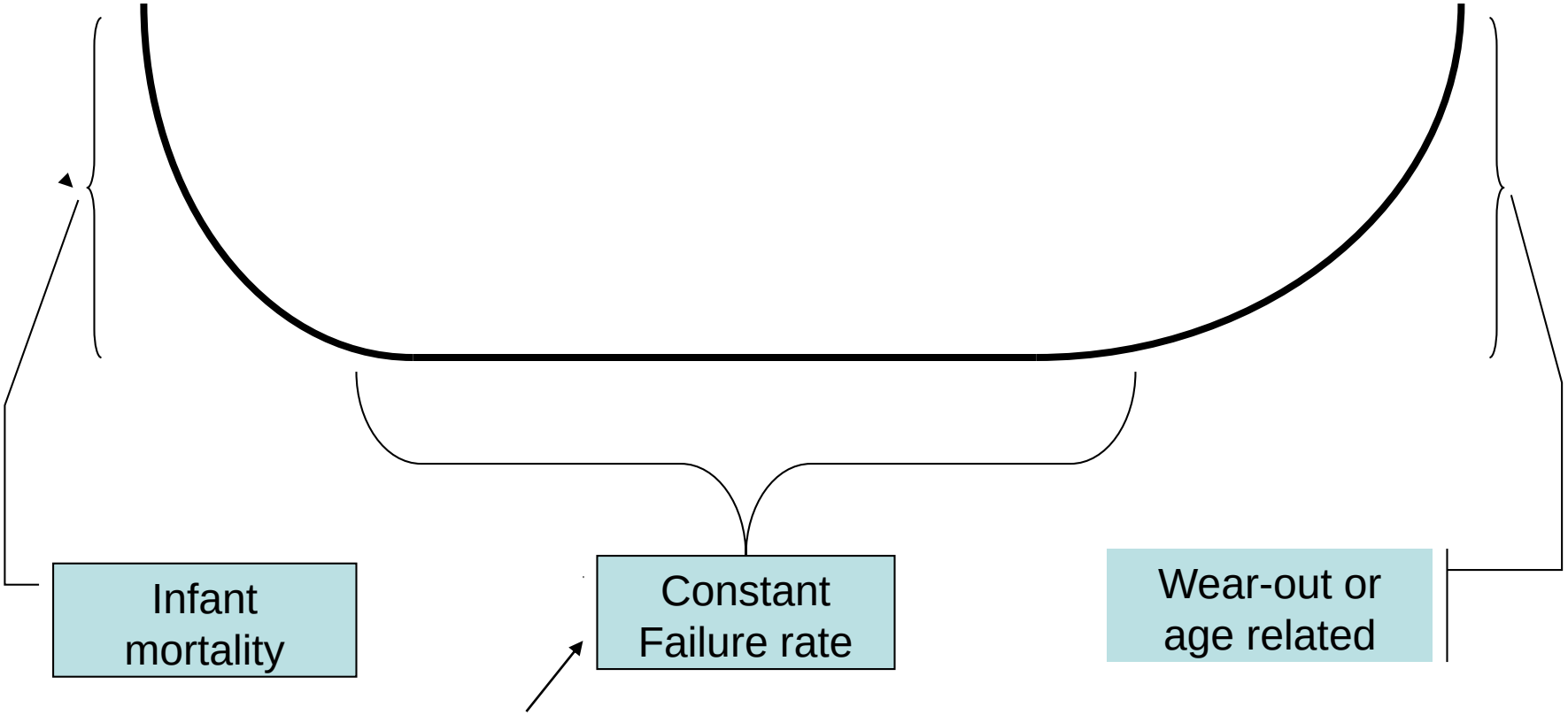
(continued)

32. **Risk Reduction Ratio (RRR):** This is another importance measure that will tell you how much a Top Event or End State will decrease if you set an items probability of failure to 0 and recalculate the end state or top event. It is equivalent to RRW.
33. **Risk Reduction Worth (RRW):** This is another importance measure that will tell you how much a Top Event or End State will decrease if you set an items probability of failure to 0 and recalculate the end state or top event. It is equivalent to RRR.
34. **Common Cause Failure (CCF):** This is a failure cause that can result in multiple failures of identical redundant equipment within a short time span therefore reducing the advantage of having redundant equipment. (e.g. contaminated lube oil fails multiple pumps in a redundant system).
35. **Big Stew (BS)** ***extra credit:*** This is a method defined by the incredibly brilliant Mark Bigler and Mike Stewart in order to model inter-phase dependencies using a linked fault tree model. The only reason Bigler is allowed to have top billing is so we can get a good and memorable Acronym (BS). It is also okay to consider the Big in "Big Stew" to be a modifier of Stew.

Basic Probability Info

Some fundamental information about different ways we use failure information

Bathtub Curve



This is where we operate as far as our model is concerned

Distribution Model is Based On

- **Our PRA model is based on the Exponential Distribution**
  - In reliability space: $P_r = e^{-\lambda t}$
  - In failure space: $P_f = 1 - e^{-\lambda t}$
  - For small values of λt, $P_f = \lambda t$ (Rare event Approximation)
  - λ is constant (i.e. we are on the bottom of the bathtub curve)

- **Do not confuse this with the uncertainty distribution that we give to λ.**

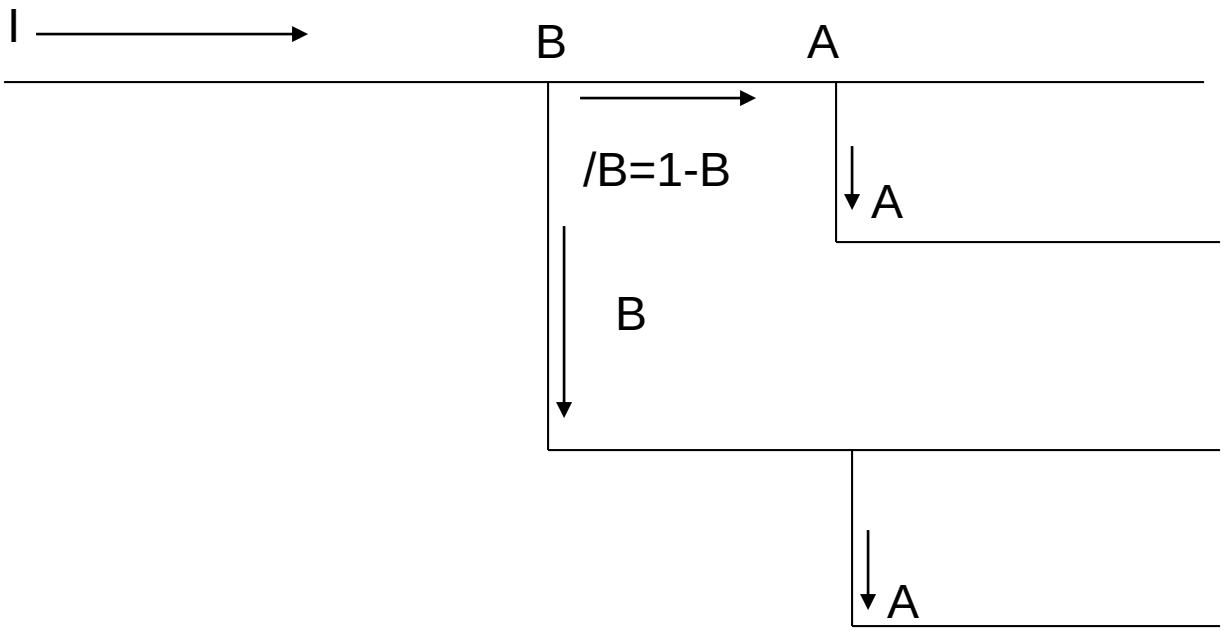
Demand Failures

- **We have discussed time rate of failures (see previous page)**
- **When items are shut down and started they need to be modeled with a failure to start, or some items fail to work when called on.**
- **These are called "Demand" failures**
- **We can use a demand failure rate we define as λ_d and can estimate a failure probability by taking this "failure rate" (I call it a rate but it is not specifically a rate of time but a rate of demands) and multiplying it by the number of demands (D).**
  - Probability of Failure = λ_d x D as long as this value is relatively small
  - We can write an equation similar to a time rate of failure probability:
    - > $P_{fd} = 1 - e^{-\lambda_d D}$
- **HRA, valves failing to open on demand or close on demand, or motors failing to start on demand etc. are demand failures and should be modeled with demand failure rates not time failure rates. In many cases a motor needs to have two failures modeled**
  - A failure to start on demand
  - A failure to continue running
- **This is true of standby equipment that is redundant that is not running and needs to be started to fulfill its safety function.**

Conditional Probabilities

- **Based on a condition that has been established (B) what is the probability of a particular event (A) happening or Given B what is the probability of A**
  - Written as P(AIB)
- **Example: Given that a tire has blown what is the probability that the landing gear will collapse?**
- **In principle the probabilities given in succeeding nodes on a path through an event tree are conditioned on what has happened before.**
  - So a node could have different probabilities based on what has happened prior or which path it is on.

Conditional Probability Continued



/B= 1-B and is called the “compliment” of B and can be written in different formats

Frequencies vs Probabilities

- **Technically you could argue probabilities are frequencies although they are defined as dimensionless (also a probability has to be between 0 and 1 and a frequency can be larger than 1)**
  - > That is we need to insure if our initiator is Probability of failure per six months of operation (this is a frequency) that the mission time in our probability calcs for the pivital events is done for six months
  - We have a failure rate (a frequency) that is multiplied by a time period (mission time) and if we use rare event we get the following equation:
    - > If lambda = 1E-5 per hour and mission time =1000 hours then the probability = 1E-2 (or Lamda x mission time)
    - > However, we still need to remember that this is the probability (a dimensionless number) for an event happening in a 1000 hour time frame
- **For Space Station we always do our calcs for a mission time**
- **Even demand failures are a rate of sorts (failure per demand). The number of demands is dictated by the number of demands that are expected per cycle or per six month period of time etc.**
- **Typically the front of an event tree (the initiator) is a Frequency (that is why it is treated differently in SAPHIRE. Probabilities all have to be between 0 and 1 a frequency does not.**
  - We could have a frequency of initiation of 10 losses of a system per year in some analysis. If this frequency is small (less than one) it can often times be treated like a probability but it still carries a per hour or per demand etc. value
- **In practice we often use probabilities and frequencies interchangeably and as long was we keep track of what we mean it is okay (probably careless and sloppy) but we can't confuse them.**
  - By definition the outcome or endstate ends up being a frequency (the intiator which is a frequency times all the pivital events which are probabilities.
  - So when we do our event trees we need to insure the mission time matches what our initiator frequency is

Quick and Dirty Calcs

- **When you need to do a calc fast (in a meeting or to check a more major calc)**
  - Use rare event if appropriate (that is the time rate of failure or the demand rate of failure times their respective mission times or number of demands do not exceed ten percent)
    - > Even here to do a quick check or sanity check using rare event will give you a conservative upper bound even if you exceed the 10% value
  - Sometimes it is easier to do the calc in reliability space than in failure space and then convert back
    - > Remember

      probability of failure = 1 - probability of success

Examples of easy calcs

- **Probability of failure of tethering is 1E-3 per tether attempt, there are 400 estimated tethers in the next 5 years. What is the probability over 5 years that we fail to tether?**
  - Build a fault tree with 400 basic events of failure to tether going through an “or” gate (not easy)
  - Solve using a binomial distribution (not easy, for me anyway)
  - Solve using rare event: 400 x 1E-3 = 0.4 (this is above the 10% value for use of rare event but gives a conservative upper bound estimate)
  - Solve using success space probability of success is

    $(1 - 0.001)^{400} = 0.67$ so probability of failure is 1 - 0.67 = 0.33
  - Use $1 - e^{-\lambda D}$, where lambda is demand failure rate to get 0.33

Data Analysis

DATA ANALYSIS

TYPES OF DATA THAT EXIST IN THE MODELS

- **Functional – A functional failure event is generally defined as failure of a component type, such as a valve or pump, to perform its intended function. Functional failures are specified by a component type (e.g., motor pump) and by a failure mode for the component type (e.g., fails to start). Functional failures are generally defined at the major component level such as Line Replaceable Unit (LRU) or Shop Replaceable Unit (SRU). Functional failures typically fall into two categories, time-based and demand-based. Bayesian update as Shuttle specific data becomes available.**

- **Phenomenological – Phenomenological events include non-functional events that are not solely based on equipment performance but on complex interactions between systems and their environment or other external factors or events. Phenomenological events can cover a broad range of failure scenarios, including leaks of flammable/explosive fluids, engine burn through, overpressurization, ascent debris, structural failure, and other similar situations.**

- **Human – Three types of human errors are generally included in fault trees: pre-initiating event, initiating event (or human-induced initiators), and post-initiating event interactions.**

- **Common Cause – Common Cause Failures (CCFs) are multiple failures of similar components within a system that occur within a specified period of time due to a shared cause.**

- **Conditional – A probability that is conditional upon another event, i.e. given that an event has already happened what is the probability that successive events will fail**

FUNCTIONAL DATA ANALYSIS (2)

DATA SOURCES

- NASA's PRACA databases are sources for Shuttle specific failure data
- Prime contractor data, when available
- Non-electric Part Reliability Database (NPRD) is a generic data source for run time failure data for mechanical components
- Electric Parts Reliability Data (EPRD) is a generic data source for run time failure data for electrical components
- Nuclear Computerized Library for Assessing Reactor Reliability (NUCLARR) is a generic data source for on demand failures
- Expert Opinion
- Miscellaneous references

FUNCTIONAL DATA ANALYSIS (4)

BAYESIAN UPDATING OF FUNCTIONAL FAILURES

- **What?**
  - It is a recognized, and standard, practice for functional failures
  - Utilizes generic databases
  - Applies a statistical technique to allow Shuttle data to update the generic values
- **Why?**
  - Provides a tool to utilize sparse data from the Shuttle to generate more accurate estimates of failure rates
  - Provides a less conservative way to estimate failure rates for components with zero failures
- **Inputs**
  - Total hours of operation or number of demands for a component
  - Number of failures experienced (derived from CAR screening and input from Engineers)

FUNCTIONAL DATA ANALYSIS (5)

BAYESIAN UPDATING

- **Performed on risk significant components**
  - List of risk significant components from iteration 2.2 of the Shuttle PRA
    - > Since the list was based on prior model there can be some components that show up as significant in iteration 3.0 that have not been screened. These will be screened for iteration 3.1.
  - Components in the top 99% or with RAW greater than 1.1 (RAW measures the change if the component failure is set to 1.0 in the model)

- **CARs were screened from first flight until 12/31/2005**

- **Only considered KSC and in flight failures**
  - Vender failures were screened out due to inability to capture corresponding operating/demand data

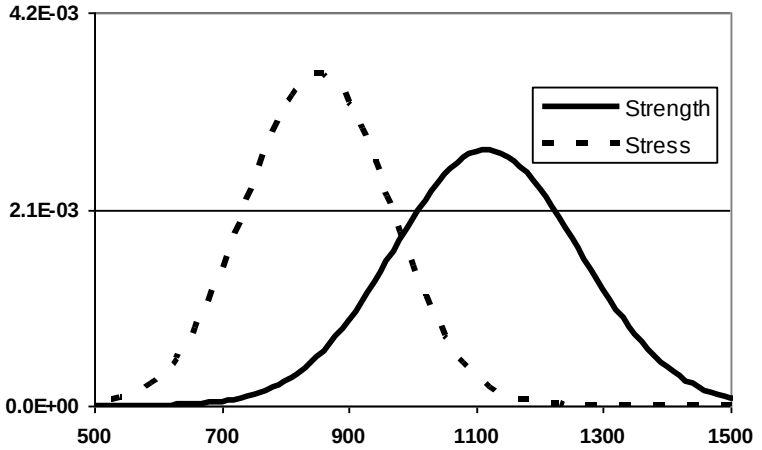
- **Partial failures were included only if there were no hard failures and were assigned either a 0.5 or a 0.1 value depending upon the severity of the failure**
  - These values came from NUREG/CR-6268, Volume 3
  - 0.5 was assigned if the component would have been capable of performing some portion of the safety function and was only partially degraded.
  - 0.1 was assigned if the component was only slightly degraded.

- **Failures were discounted based upon corrective action**
  - If sufficient information was available the “fix factor” was calculated by taking the failure rate before the fix and dividing by the failure rate after the fix
  - If sufficient information was not available the “fix factor” was assumed to be one of the following depending upon the type of corrective action
    - > 50% for design changes that were described as “improvements” or procedural changes
    - > 90% for design changes that “eliminated” the failure mode

PHENOMENOLOGICAL DATA DEVELOPMENT (2)

SPLAT (SHUTTLE PRA LEAK ANALYSIS TOOL)

SPLAT calculates the probably of a leak occurring, then determines the probability that the leak exceeds the critical leak size. It is a standard stress-strength model and where leaks are stresses and the critical leak size is the strength.



Inputs are entered as distribution parameters and results are calculated using Monte Carlo sampling.

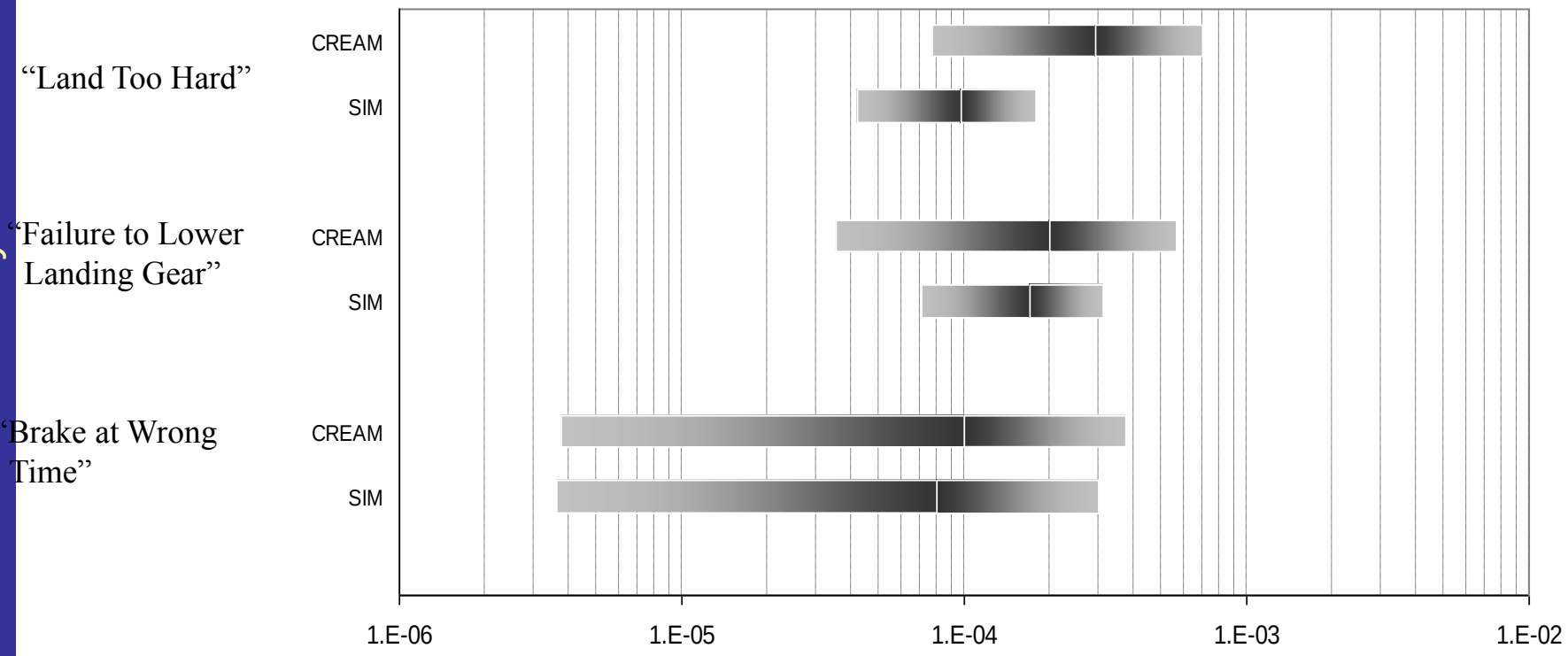
Leak Probability	Critical Leak Size	Mission Leak Size
Exponential	Exponential	Exponential
Lognormal	Lognormal	Lognormal
Normal	Normal	Normal
Gamma	Gumbel	
Point Estimate	Uniform	
	Point Estimate	

HUMAN RELIABILTY ANALYSIS (HRA) DATA DEVELOPMENT

- **HRA is a method used to describe, qualitatively and quantitatively, the occurrence of human failures in the operation of complex machines that affect availability and reliability.**
- **Modeling human actions with their corresponding failure in a PRA provides a more complete picture of the risk and risk contributions.**
- **A high quality HRA can provide valuable information on potential areas for improvement, including training, procedural and equipment design.**
- **Screening analysis is performed on the bulk of the human errors with a detailed analysis only performed on the significant contributors**
- **There are Many Different Methodologies for Model Human Errors in PRA**
  - For the Shuttle PRA Cognitive Reliability and Error Analysis Method (CREAM) was selected as the primary method for detailed analysis
    - > It was selected as one of the NASA recommended HRA techniques
  - The results from CREAM have been favorably benchmarked against other methodologies and simulator data as part of the Shuttle PRA
  - The majority of HRA events are processed with a screening analysis that is essentially based on the Technique for Human Error Reliability Prediction (THERP) in NUREG/CR-1278. THERP is a recognized HRA technique that has been used for over 20 years, primarily in calculating Human Error Probability (HEP) in nuclear power plant PRAs.
    - > The screening table was easy to apply and gave conservative values. If an HRA event that was developed using the screening table became a significant contributor it was then re-modeled using CREAM

HRA

Comparison of Simulation Data and CREAM Results



The Cream results have since been Bayesian updated using the simulator data

COMMON CAUSE DEFINITION

- **In PRA, Common Cause Failures (CCFs) are failures of two or more components, subsystems, or structures due to a single specific event which bypassed or invalidated redundancy or independence at the same time, or in a relatively short interval like within a single mission**
  - May be the result of a design error, installation error, or maintenance error, or due to some adverse common environment
  - Sometimes called a generic failure.
- **Common Cause, as used in PRA, is <u>not</u> a single failure that takes out multiple components such as a common power supply to computers or common fluid header to multiple pumps.**
  - Single point failures such as these are modeled explicitly in a PRA

COMMON CAUSE MODELING

(More details and examples on this later)

- **All large PRAs of complex and redundant machines must include "common cause" effects to be complete and accurate**

- **Common Cause are those conditions that defeat the benefits of redundancy**
  - Not "single point failures"
  - Similar to "generic cause"

- **There are three recognized ways to perform common cause modeling:**
  - The Beta Model
  - The Multiple Greek Letter Model
  - The Alpha Model

- **We use an iterative approach to modeling common cause first the Beta Model approach is used and if it shows up as a risk driver a Multiple Greek Letter Model is used**

- **Generic data from NUREG/CR-5485 for the majority of the events since there are few cases where there is enough Shuttle data to develop Shuttle specific values**
  - RCS Thrusters and ECO sensors are examples of cases where Shuttle specific data is used to calculate the common cause parameters

COMMON CAUSE MODELING (2)

HOW THE BETA MODEL APPROACH WORKS

- **Susceptibility groups (groupings of similar or identical equipment) of redundant trains or components are identified**
- **A common cause basic event is defined for these groups**
- **The common cause basic event failure rate is generated by taking the independent failure rate times a "Beta" factor.**
  - For the beta model it does not matter how many components are in the group
  - The "Beta" factor represents the probability of 2 or more failures given a failure has occurred
    - For this reason, the Beta Model may be conservative for component groups larger than 2.
- **The "Beta" factor is taken from NUREG/CR-5485 and has a different value for "Operating" failures vs. "Demand" failures**
  - Operating failures the "Beta" value is 0.0235
  - Demand failures the "Beta" value is 0.047

COMMON CAUSE MODELING (3)

HOW THE MULTIPLE GREEK MODEL APPROACH WORKS

- Similar to the Beta Model except that the Multiple Greek Model takes credit for the full redundancy and therefore can be much more complicated
  - For a 3 component group, there is a “beta” factor and a “gamma” factor where the “beta factor is still the probability of 2 or more failures and the “gamma” factor is the probability of 3 or more failures given 2 or more failures.

CONDITIONAL PROBABILITY

- **Given that an event has already happened what is the probability that successive events will fail**
  - Example : Given two blown tires in the time interval between main gear touch down and nose gear touch down what is the probability that the Orbiter crashes (i.e. strut fails or crew looses control of vehicle)

- **Conditional probabilities are typically relatively large (e.g. values like 0.1 to 0.9) and are usually derived from expert opinion or direct experience.**

CONCLUSIONS

- **Like redundancy helps but may not help as much as you think because there is a point of diminishing returns with like redundancy**
- **Redundant but diverse designs can defeat common cause and supply the best reliability**
- **Failure to model common cause will lead to underestimation of the risk**
- **Common cause parameters based on real data are hard to derive due to a lack of data**
- **A high common cause parameter does not mean that a component is unreliable, it just means that given that one component has failed, additional similar components are more likely to fail**

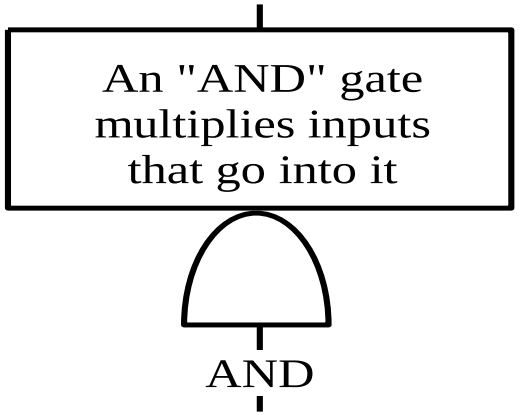
Reading a Fault tree
(A Very Basic Explanation)

Fault trees are often used to perform Probabilistic Risk Assessments (PRA). A basic understanding of how to read a fault tree is needed. The following few slides describe a **few** of the most commonly used symbols used to build fault trees and gives a very basic example. The symbols shown in this document are specific to the SAPHIRE computer program but generally conforms to most fault tree symbols. In some cases the symbols are demonstrated by using the “Graphic” editor symbols in SAPHIRE and in some cases the “Logic” editor symbols are used.

Fault Trees

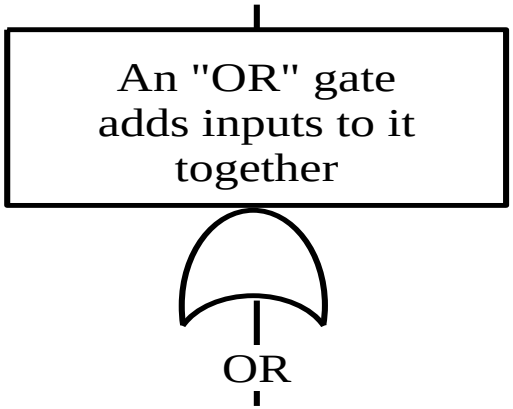
- **Used to calculate probability of failure**
- **Examples of Fault trees developed for Shuttle systems:**
  - Electrical Power
  - Auxiliary Power Unit
  - Hydraulics
  - ECLSS
    - > Etc.
- **Includes hardware, software, human errors,**
- **Includes common cause failures**
- **Fault trees show interdependencies among distributed systems by including the interactions with all supporting equipment**
  - MDMs
  - Coldplates
  - RPCMs / DDCUs
  - Environmental controls

- **The “And” gate:**
  - Fault trees use Boolean Algebra to solve models that are built. The “and” gate takes whatever probabilities that are in-put to it and multiplies them together.



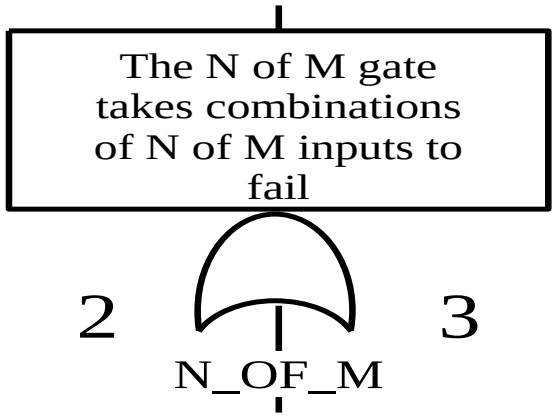
The “Or” gate:

- The “or” gate takes whatever probabilities that are put into it and adds them. In Boolean algebra the adding is a little more involved. If the probabilities of A or B, are put into the “or” gate the algebraic equation is A + B – A x B. If the probabilities are low (i.e. less than .1) then the answer can be approximated by just A + B (also known as the “rare event approximation”)



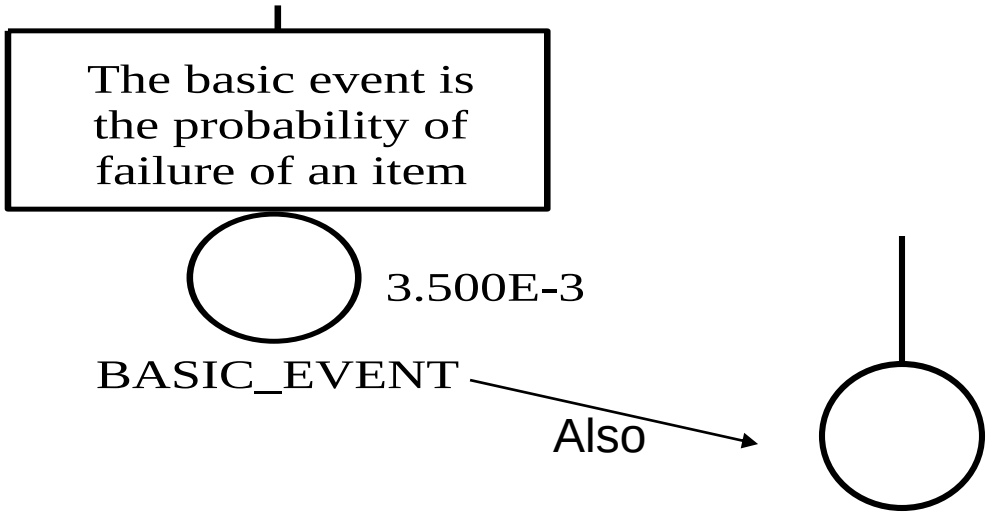
The “N of M” gate:

- The “N of M” gate is used to define combinations of “M” things taken “N” at a time. This same logic can be defined using combinations of “and” and “or” gates. The “N of M” gate is a shorthand for doing this. An example would be to take three items “A, B and C” two at a time to get the following: AB, AC, and BC.



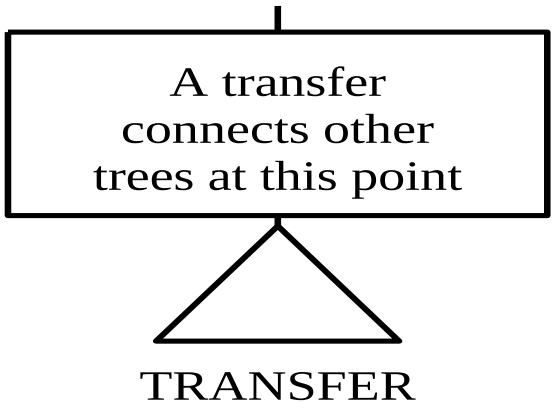
The "Basic Event":

- The "basic event" is the item whose probability is modeled. This is the most basic (lowest) level that we model to. There is a tendency to model down to too low a level of detail. However, it is a mistake to model down to a lower level than data can be acquired to represent the failure probabilities for that item.



The "Transfer gate":

- The "transfer gate" is used to connect parts of fault trees together. It is used to split the tree up so that the pieces can be fit onto a single piece of paper to be more easily printed out and read or also if several fault trees use the same equipment then the transfer can be used to model that equipment once to be used in many different places in other trees.



Some other Symbols you might use or see

Undeveloped Event

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

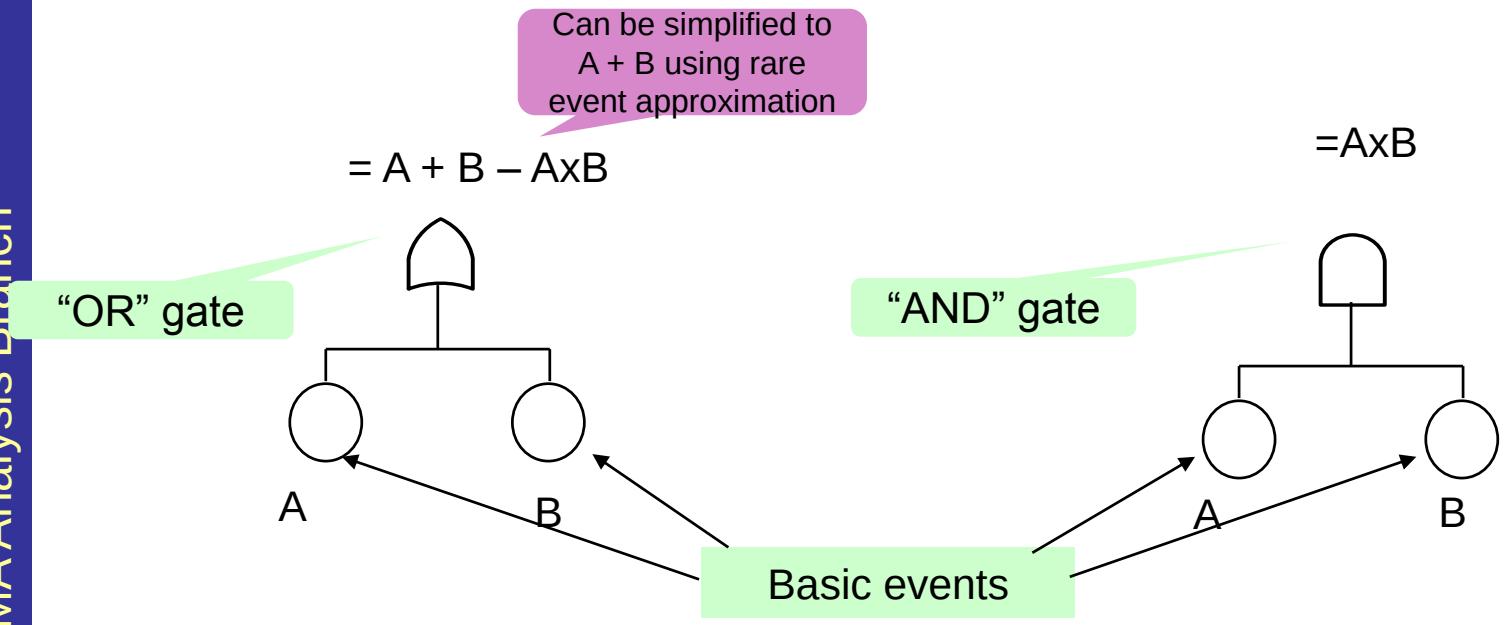
House Event

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

Undeveloped Transfer

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

Some Fault Tree Basics



Boolean algebraic identities
(Just a few basic ones are given)

Additive Identities:

$A + 0 = A$

$A + 1 = 1$

$A + A = A$

$A + \overline{A} = 1$

Multiplicative Identities*:

$0A = 0$

$1A = A$

$AA = A$

$A\overline{A} = 0$

$A = A + AB$

**Note: Multiplication (Logical AND) is implied when two variables are written next to each other.*

- **Example:**
  - To demonstrate the use of the symbols to model a system an example fault tree is done and represented in Figures 1 and 2. Figure 1 is the fault tree for system “Station” and figure 2 is a piece of the fault tree that is modeled separately and is connected by a “Transfer” gate (the “transfer” gate name must be the same name as the top of a tree that is being transferred). In Figures 3 and 4 we find the same set of logic represented using the “Logic” editor portion of SAPHIRE. The logic editor graphics give a more compact version of the logic and is sometimes preferable to use since it reduces the number of pages needed to represent the tree.

Figure 1.

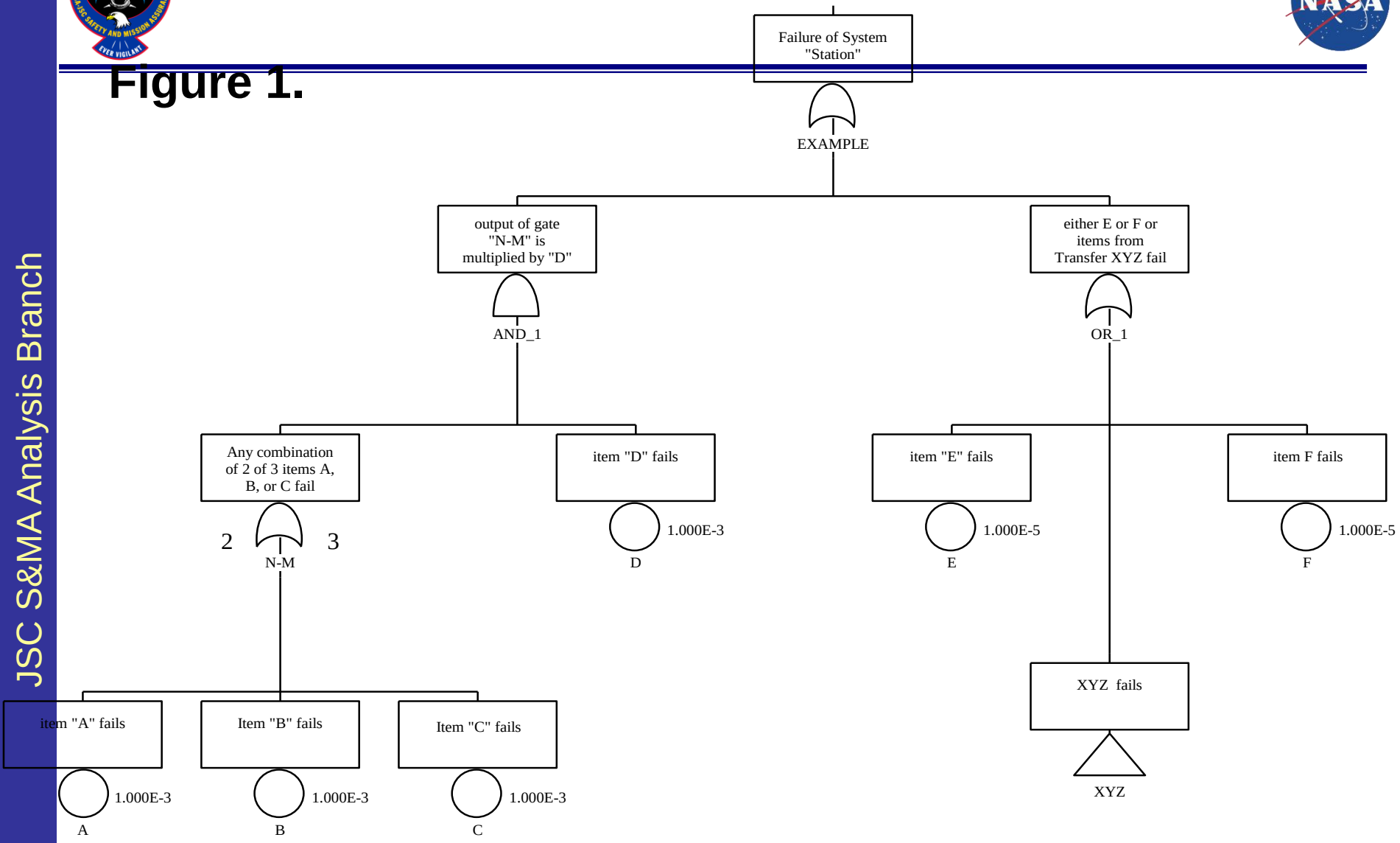

Failure of System "Station"
EXAMPLE
output of gate "N-M" is multiplied by "D"
AND_1
either E or F or items from Transfer XYZ fail
OR_1
Any combination of 2 of 3 items A, B, or C fail
2
3
N-M
item "D" fails
1.000E-3
D
item "E" fails
1.000E-5
E
item F fails
1.000E-5
F
item "A" fails
1.000E-3
A
Item "B" fails
1.000E-3
B
Item "C" fails
1.000E-3
C
XYZ fails
XYZ

Figure 2.

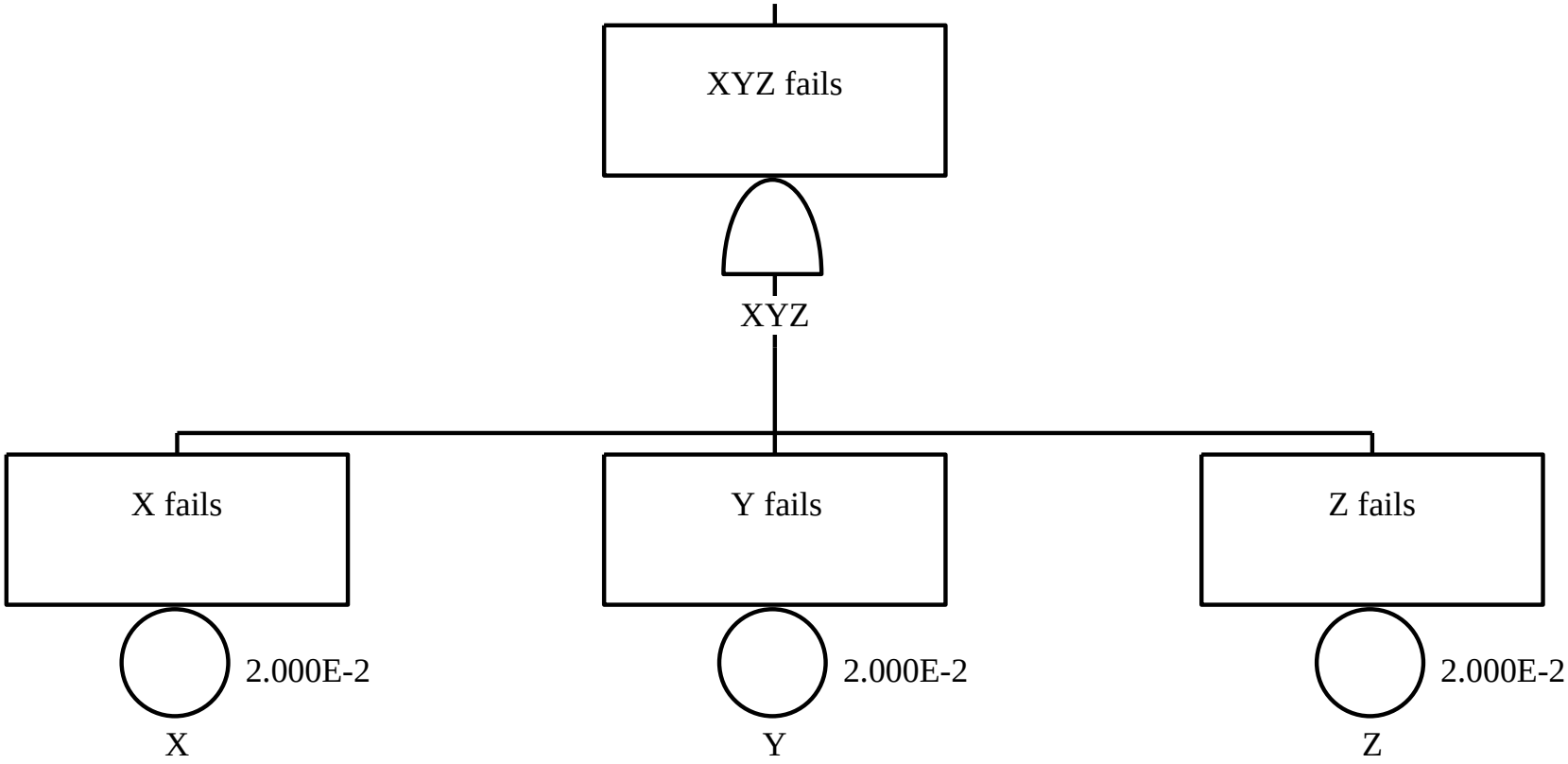

XYZ fails
XYZ
X fails
Y fails
Z fails
2.000E-2
2.000E-2
2.000E-2
X
Y
Z

Figure 3.

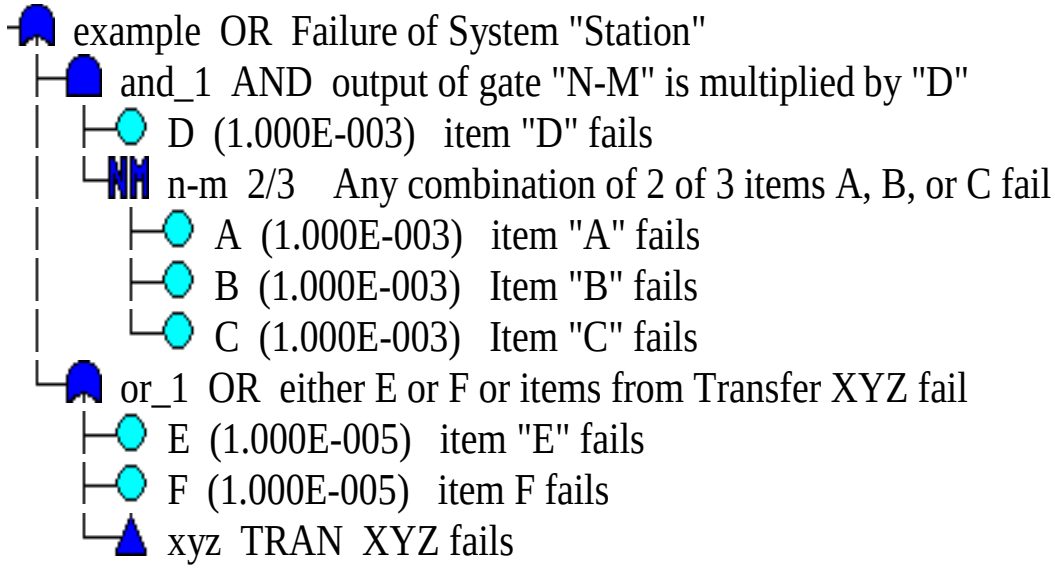
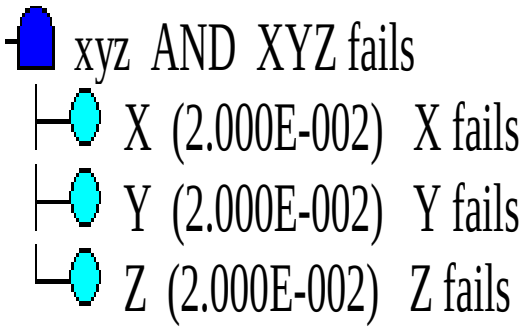


Figure 4.



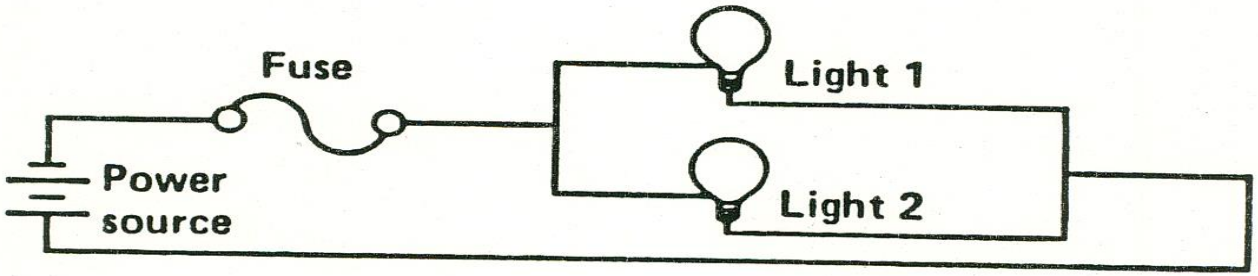
Cut Sets

Cut No.	% Total	% Cut Set	Prob./Frequency	Basic Event	Description	Event Prob.
1	35.71	35.71	1.000E-005	E	item "E" fails	1.000E-005
2	71.42	35.71	1.000E-005	F	item F fails	1.000E-005
3	99.99	28.57	8.000E-006	X	X fails	2.000E-002
				Y	Y fails	2.000E-002
				Z	Z fails	2.000E-002
4	99.99	0.00	1.000E-009	A	item "A" fails	1.000E-003
				B	Item "B" fails	1.000E-003
				D	item "D" fails	1.000E-003
5	99.99	0.00	1.000E-009	A	item "A" fails	1.000E-003
				C	Item "C" fails	1.000E-003
				D	item "D" fails	1.000E-003
6	99.99	0.00	1.000E-009	B	Item "B" fails	1.000E-003
				D	item "D" fails	1.000E-003
				C	Item "C" fails	1.000E-003
		Grand total~	**2.8E-5**			

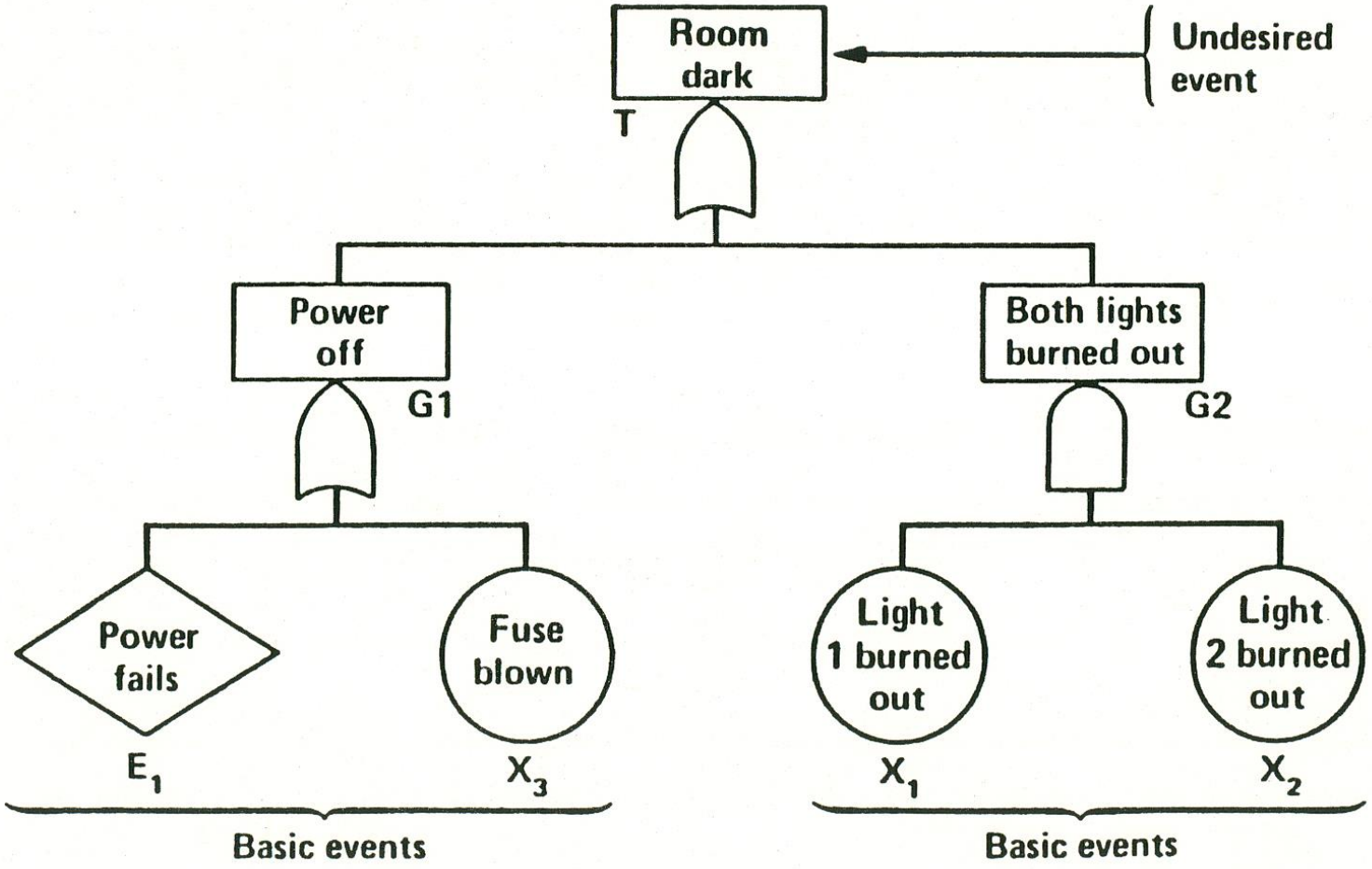
In this example we did not consider common cause. More about that later.

Simple System Fault Trees and Minimal Cutset Problems

DRAW A FAULT TREE FOR THE SYSTEM BELOW. THE TOP EVENT OF THE FAULT TREE IS "ROOM DARK"



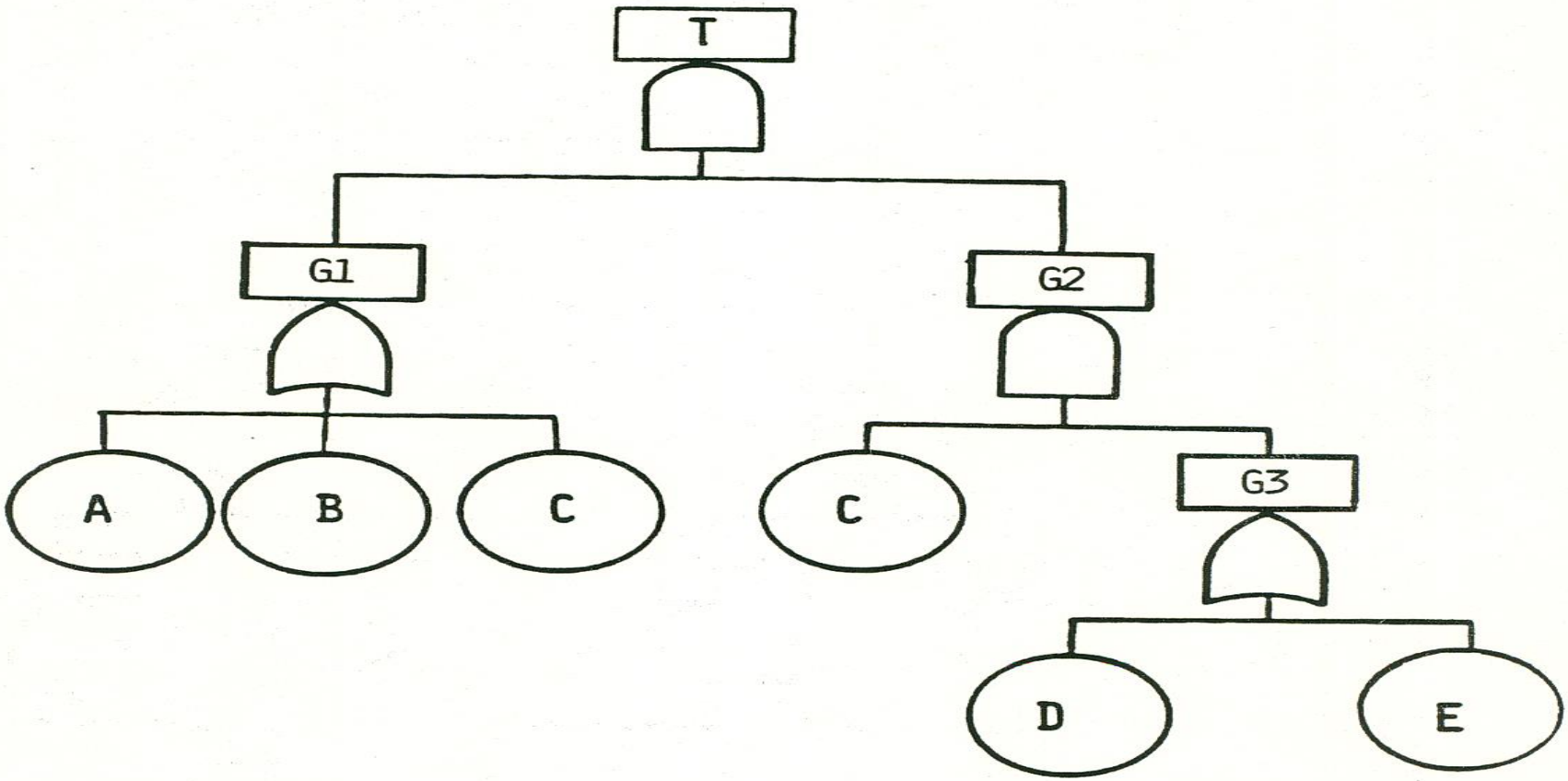
A SOLUTION FAULT TREE FOR PROBLEM



The Cut set Solution to the Model

Cut No.	% Total	% Cut Set	Prob./Frequency	Basic Event	Description	Event Prob.
1	82.04	82.04	5.000E-003	E1	Power fails	5.000E-003
2	98.45	16.41	1.000E-003	X3	Fuze blown	1.000E-003
3	100.00	1.64	1.000E-004	X1	Light 1 burned out	1.000E-002
				X2	Light 2 burned out	1.000E-002

EXAMPLE OF MINIMAL CUT SET GENERATION



GENERATION OF THE MINIMAL CUT SETS FROM A FAULT TREE REQUIRES FOUR STEPS

STEP 1 — GENERATE THE INTERMEDIATE EVENT EQUATIONS FOR THE FAULT TREE

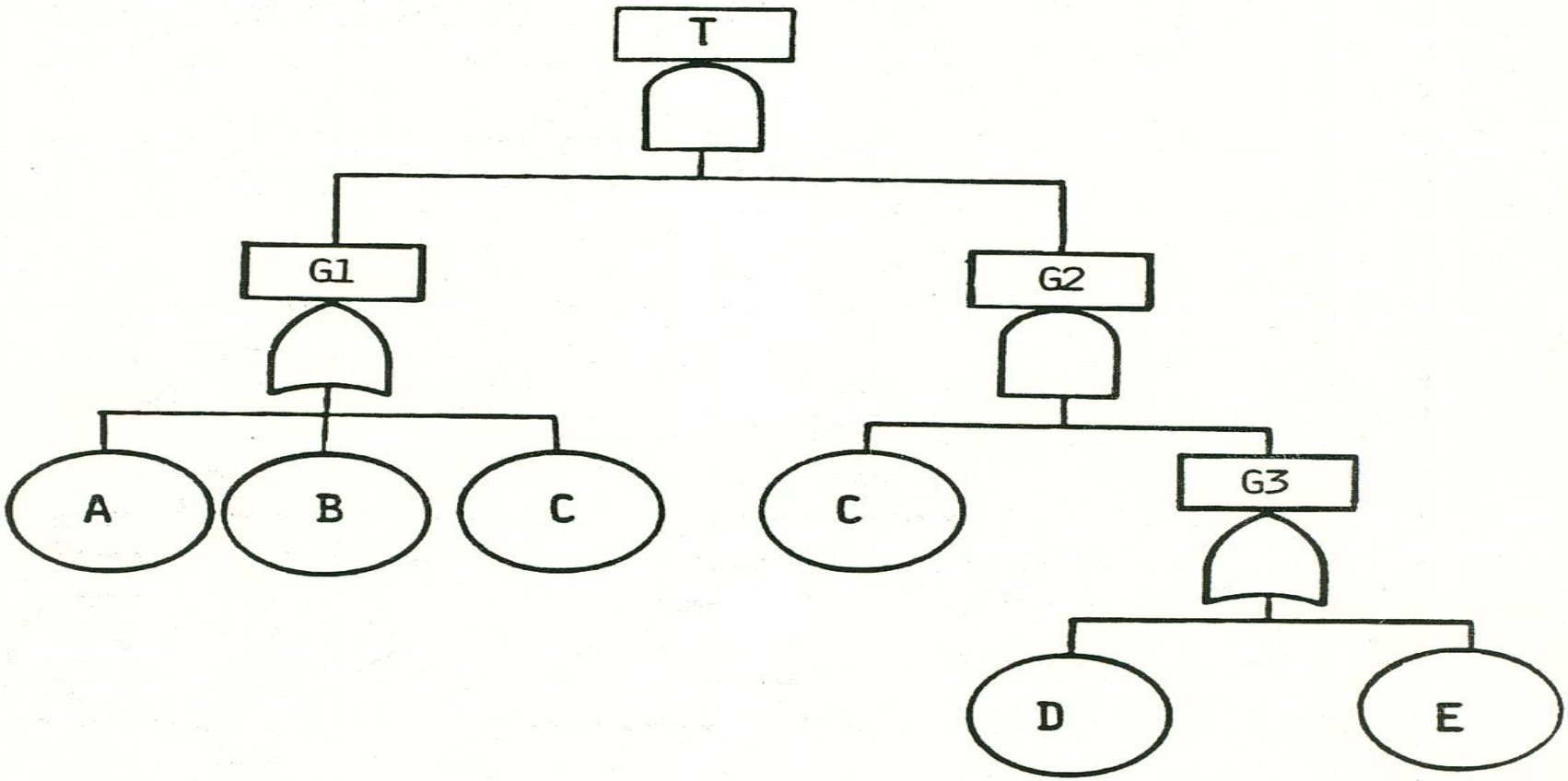
STEP 2 — GENERATE AN EQUATION FOR THE TOP EVENT THAT IS A FUNCTION OF ONLY BASIC EVENTS

STEP 3 — REDUCE THE EQUATION GENERATED IN STEP 2 BY THE BOOLEAN LAWS OF ABSORPTION

- $P \bullet P = P$
- $P + P \bullet Q = P$

STEP 4 — WRITE THE EQUATION GENERATED IN STEP 3 IN A SUM-OF-PRODUCTS FORM

EXAMPLE OF MINIMAL CUT SET GENERATION



CUT SETS

1) A, C, D
2) B, C, D
3) C, C, D
4) A, C. E
5) B, C, E
6) C, C, E

MINIMAL CUT SETS

1) C, D
2) C, E