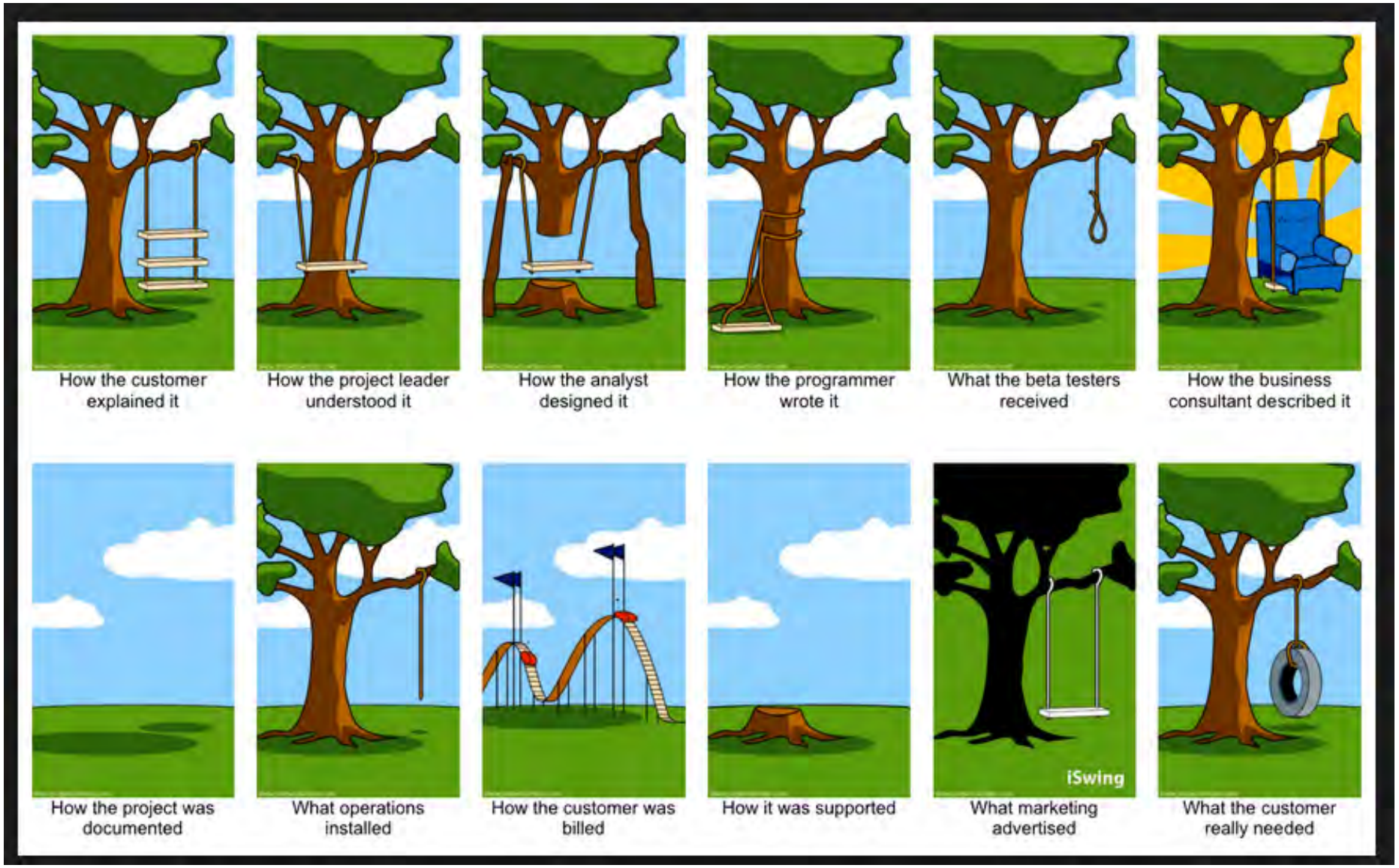


Requirements should always be analysed to reconcile the user's actual problem with the end solution. The rest can only be avoided if the requirements are good.

Effect of project size and complexity

Aspect	Small Projects	Large Projects
Specifications	Minimal	Extensive
Status Information	All in one person's head	Distributed across a design team
Design Information	All in one place	Distributed across design teams and piles of documents
Sources of Error	Faulty/Poor specifications Design errors	Faulty/Poor specifications Interface errors Inconsistent Assumptions Inconsistent definitions Sourcing problems Design errors

Design errors are much quicker and easier to fix than a flaw due to a poor requirement

Requirements

- A singular documented physical or functional need that a particular artefact aims to satisfy
- A positive statement specifying an attribute or capability which is verifiably present in the final artefact
- Does not include design decisions
- *“The inputs shall be reviewed for adequacy. Requirements shall be complete, unambiguous and not in conflict with each other”* – ISO9001:2008
- *“Requirements mature like good wine*”, - Robert Halligan (Systems Engineer)

Examples of BAD requirements

- “The CSCI SRA phase will provide performance requirements for high, medium and low system loads”
- The xxx CSCI shall be designed and coded in such a manner as to make efficient use of available computer and communications resources.
- The short response time is based on a nominally 20ms up-line transmission time, 30ms down-line transmission time and 20ms processing time.
- Processor performance equivalent to Intel 80386 or Motorola 68030 is considered adequate.

Good requirements specify…

- Functions (what the artefact does)
- States and modes (sets of conditions and functional groups)
- Performance (how well a function is performed)
- External interfaces (the required characteristics at a point in the system where it interacts with the world or next higher assembly)
- Environmental (limits the effect that the artefact has on the environment, or vice versa.)
- Resources (limits the usage or consumption of an externally supplied resource)
- Physical (limits physical properties of the artefact or entire system.)
- Limits can be minimum (average speed over the course must exceed …) or maximum (the vehicle shall not exceed the microlight weight class limit.)

State/Mode requirement

- Defines alternative sets of characteristics that an article must possess, but not simultaneously.
- Examples:
  - The maintenance service shall have two modes:
    a) Normal response mode
    b) Emergency response mode
  - The radar shall operate as part of the xxx network (network mode) and shall also have an autonomous mode of operation.
  - The system shall be able to operate in a degraded performance mode if it sustains damage.

Functional Requirements

- States *what* a system will *do*
- Must include an action and an object of action
- Often requires qualifiers (accuracy, resolution, conditions, etc)
- Examples
  - The device shall display the local time
  - The fuel feed system shall provide cooling to the exhaust gas expansion system

Performance Requirements

- States how well a function should be performed
- Relate only to functions
- Stated only in quantitative terms
- Always best to include performance within a corresponding functional requirement
- Examples
  - The error of computation of impulse response of the oblique ionospheric path in a bandwidth of 1kHz at any frequency between 3 and 30MHz shall not exceed 100ns.
  - The platform shall operate continuously at an altitude sufficient to ensure that two platforms can observe the entire controlled airspace allocated to XXX organisation.

External Interface Requirements

- Define the system boundary
- Specify everything that will enter or leave the system
- Specify the external systems that provide inputs
- Specify the external systems that accept the outputs
- Specify both normal and anomalous I/O
- may provide nomenclature for interfaces
- Example: The system shall provide standard rack mount interfaces which comply with EIA310.

Environmental Requirements

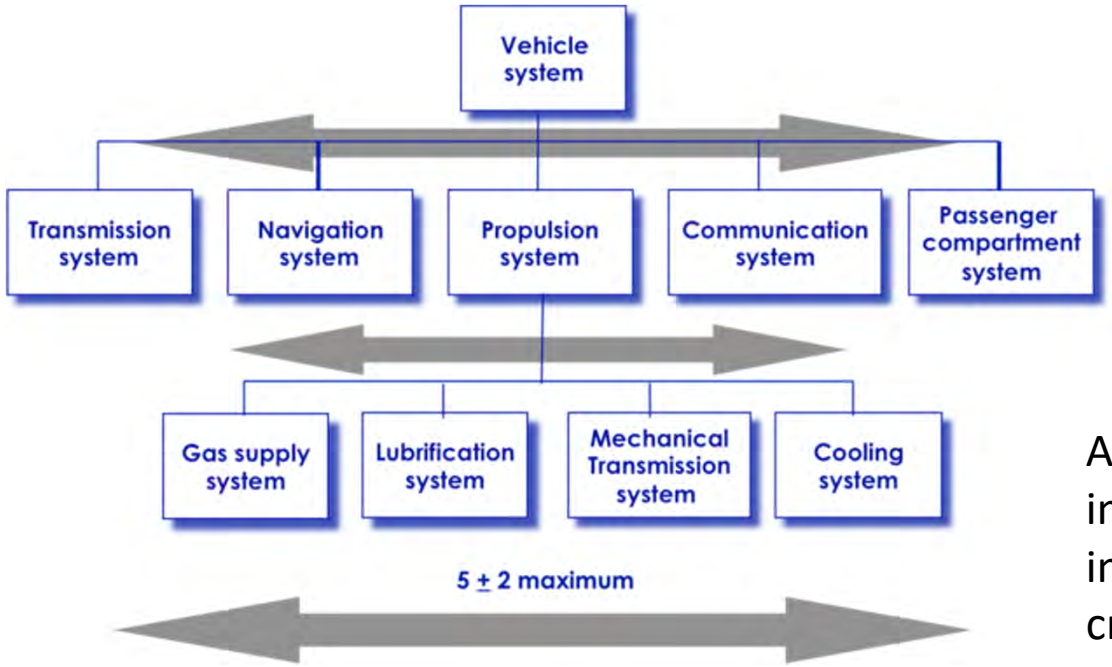
- States the external physical environment within which all other requirements are to be satisfied
- Include parameters such as dust, humidity, ambient temperature, vibration, etc
- Include rates of change
- Include constraints on the effect of the system on its external environment.
- Examples
  - The artefact shall operate in an ambient temperature range of 10 – 40 degrees C; and humidity of 20 – 90 percent, non condensing.
  - The inner containment vessel shall prevent 99.99% of all high energy neutrons from reaching the outer containment shell.

Other Requirements

- Resource requirements state limits on consumption or use of externally provided resources
  - The Electric Propulsion unit shall not draw more than 400 W
- Physical requirements state specific attributes such as mass or dimensions
  - The total volume shall not exceed 3 standard CubeSat Units
- Other qualities may also be specified such as transportability, survivability or compatibility among others.
- Important to consider what transportability means and how it can be tested.
  - The unity must be man portable within SANS18001- OH&S standards and accompanying regulations.
- Same applies for all requirements but particularly for things such as efficiency, reliability, reusability, etc.

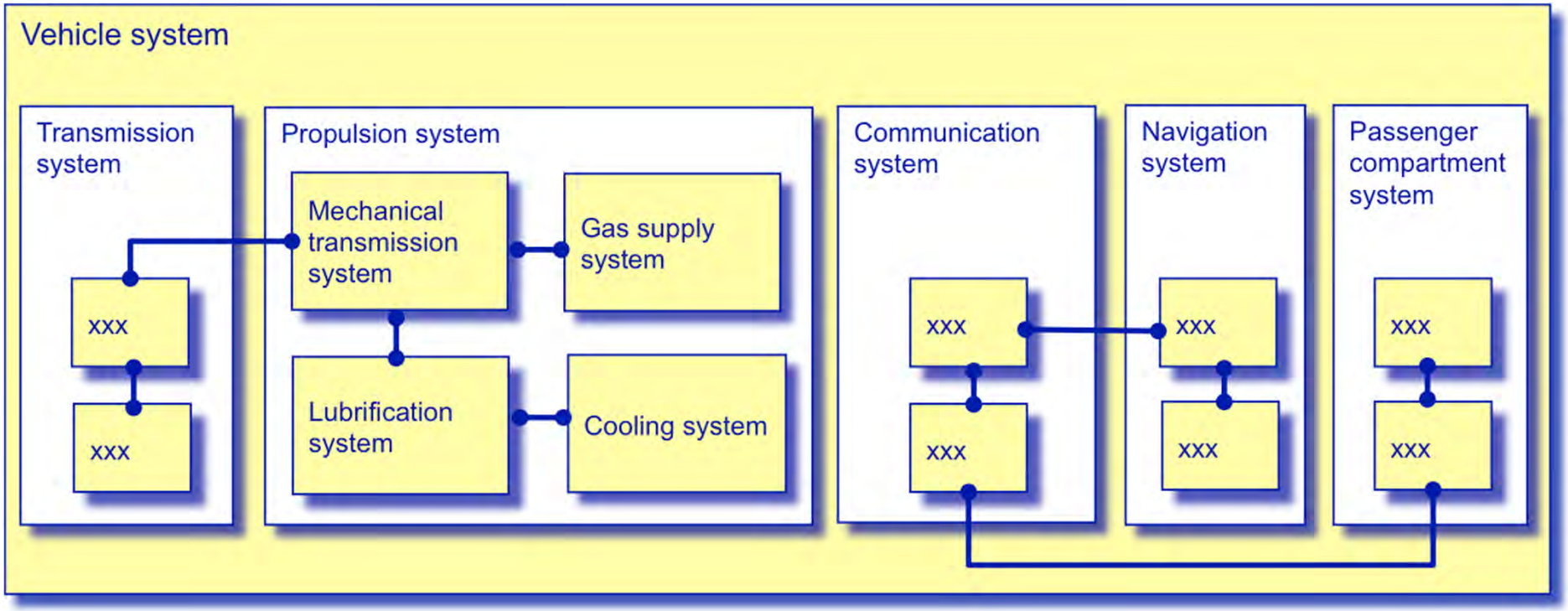
A good requirement has…

- Validity – adds value
- Correctness – is factually correct
- Consistency – does not conflict with other requirements
- Clarity – is understandable by the intended reader (err on the side of obtuseness here)
- Non-Ambiguous – the requirement is not open to interpretation
- Traceability – can be traced through design and verification
- Singularity – only action verb and one object of action
- Feasibility – it can be satisfied within the current understanding of the laws of physics.
- Non-redundant – no duplication within a set of requirements
- No design or process decisions - Does not make design decisions (some exceptions may apply)
- Function oriented – States problems, never solutions. This applies to the set of requirements as a whole.



Architectural Hierarchy Diagrams

A Physical Model of the system should include all interfaces and should track all information, energy or matter that crosses a system boundary.



Story so far

- Looked at tools to help identify the actual user problem
  - Context diagrams to understand all the external influences
  - 9 windows diagrams to plot the problem history and build use case senarios
  - System diagrams to identify all information, energy and matter that crosses he system interface (where it comes from and where it goes as well
- always interrogate what the user says he wants in order to understand what the problem is
- Have developed a set of guidelines on what makes for good requirements
- Assumed you know how to turn the requirements into an artefact

Making your design reliable

- Reliability is not something that can be easily added to a design after it is complete.
- Reliability Engineering is about managing the dependability of an artefact over its life cycle.
- Reliability engineering deals with the estimation, prevention and management of high levels of *lifetime* engineering uncertainty and risks of failure.
- Dependability is the ability of an artefact or system to function understated conditions for a specified time.
- Every design should be dependable and is the sole responsibility of a design engineer.

Failure Mode and Effect Analysis

- Often combined with a criticality analysis (FMECA)
- When used with Root Cause analysis forms the foundation of Reliability Centred Maintenance (See RCMII by John Moubray)
- *Failure modes* describe the ways in which a system may be unable to fulfil any or all of its functions
- *Effect analysis* looks at the consequences of such failures focusing on severity, cause and frequency of occurrence

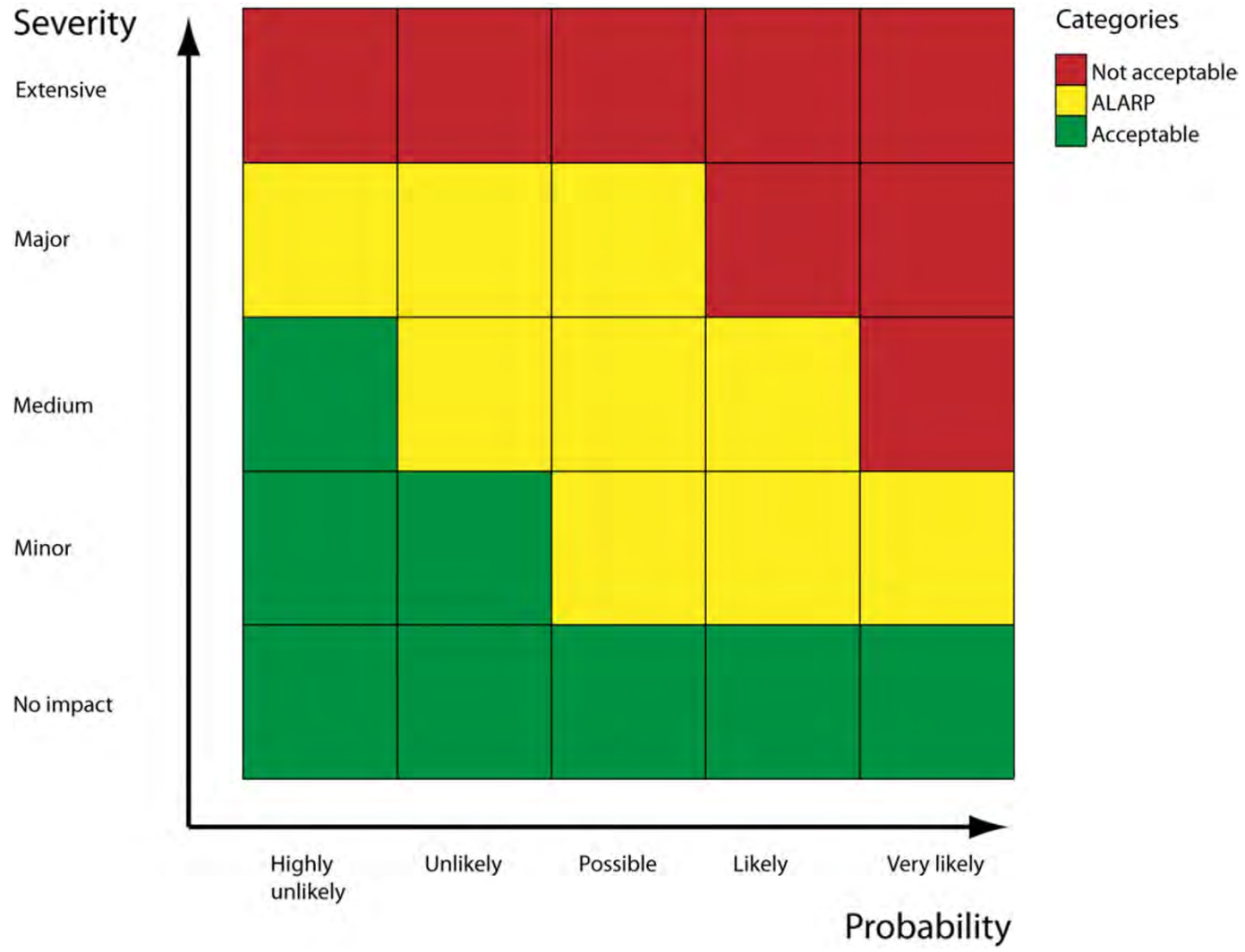
Failure modes

- The system has failed at any point where it is no longer able to deliver any or all of its stated functions at the required performance metrics
- Failure modes can include operating at a less than normal operating point (degraded operation failure)
- The cause of the failure mode needs to be understood
- Can the failure mode be detected?
- Example:
  - Oil seal on a landing gear break forms a slow leak.
  - Cause: Worn or damaged seal.
  - Detection: Loss of break fluid pressure on that hydraulic line
  - Severity: Slight leak, minor loss of fluid, degraded breaking performance; could also be significant leak, total loss of pressure and complete failure of the breaking system

FMEA

1. Define all ratings and assumptions
2. List all potential failure modes. This requires consideration of all systems and subsystems as well as all functions
3. List all the effects of each failure mode. Consider what the consequences are and who is affected.
4. Estimate the severity of the effect and rate it from minimal at the low end, to severe in the mid range and catastrophic as the worst possible outcome (loss of life would be catastrophic)
5. List the cause(s) of the failure and the frequency of occurrence
6. List means of detection and process controls
7. Determine the Risk priority – Any catastrophic outcome should score the highest possible risk, regardless of the frequency of occurrence.
8. Make recommendations to appropriately address the risk
9. Make sure that someone takes responsibility for addressing the recommendations and that the actions taken are reported back and documented.

Risk Matrix


Severity
Extensive
Major
Medium
Minor
No impact
Highly unlikely
Unlikely
Possible
Likely
Very likely
Probability
Categories
Not acceptable
ALARP
Acceptable