

## The Five “Whats” of System Safety

Donald S. Hanline II; APT Research, Inc.; Huntsville, Alabama

Saralyn Dwyer; APT Research, Inc.; Huntsville, Alabama

Keywords: requirements, lessons learned

### Abstract

Comprehensive system safety requirements are important for the System Safety Program, overall cost, schedule and acceptance of the product. Failure by System Safety Engineers to identify requirements, convey them to Systems Engineering, and ensure incorporation into requirements documentation may impair these objectives, though the design “met requirements.” Late requirements impair development, causing design and testing not originally scheduled. The Systems Safety Engineer is responsible for compliance with safety requirements, including customer and independent organizations requirements, necessary to get the product to the user. System Safety is complex, involving many discrete actions. Consider grouping actions into five logical “Whats”, which contain numerous pre-existing ideas combined with a few new ones.

What the System Safety Engineer/Program:

- 1) wants to do to assure a safe product (goals).
- 2) is contractually required to do (requirements).
- 3) may consider doing to achieve the objectives (planning).
- 4) does to perform the task (actions).
- 5) determines will be accepted as final product (deliverables).

“Correct” specification safety requirements applies knowledge of final product acceptance and usage. This paper presents a safety assessment approach incorporating not only the hardware and software comprising the system, but the functions, personnel, procedures and independent approvals required to achieve safe products.

### Introduction

The System Safety process, as it applies to product development, is fairly well defined and if well executed, fits in with the overall system engineering and product design effort. One could probably obtain near universal consensus that a proactive System Safety program, integrated into the product development process, is desired. Successful program development may benefit from an assessment of the end product, its implementation and required support functions at program onset. This paper refers to a “generic” product development from program startup to end use and highlights some of the pitfalls associated with failure to analyze the end steps of the development cycle early in the program.

The purpose of this paper is not to detail or attempt to redefine the System Safety process, but to increase awareness, through experiences, observations, mistakes and lessons learned of potential ways to improve the System Safety Engineer’s input to successful product development. The approach outlined in this paper is primarily a logical grouping and ordering of pre-existing System Safety processes. The anticipation is that as the reader digests the contents of this paper, several responses may be evoked. Those expected responses may range as follows:

- “There is nothing new here.” Author response: The reader is correct and many good ideas involve the application of existing methods.
- “I already know all of this.” Author response: The reader, as a safety professional, should know all of this.
- “This would never happen on my program because I have 236 years of safety experience.” Author response: Safe money bets everyone has at least one story that did happen to him or her. It has happened to the author repeatedly during his growth as a System Safety Engineer.
- “Hey, this makes some sense.” and “How can I use this in my efforts?” Author response: Now you get it and can begin the difficult part, which is implementing this approach.



***Thank you for your  
interest in our papers!***

*For the rest of the paper, please email  
[aptinfo@apt-research.com](mailto:aptinfo@apt-research.com)*