A-P-T RESEARCH, INC.
AN EMPLOYEE-OWNED COMPANY

A System Safety Process for Artificial Intelligence and Machine Learning Based Products

Assessing the safety of a system with nondeterministic components

Prepared by:
A-P-T Research, Inc.
4950 Research Dr.
Huntsville, AL 35805
Phone: 256.327.4015
Fax: 256.837.7786
www.apt-research.com
5 November 2021
T-21-00600

1.0 Introduction

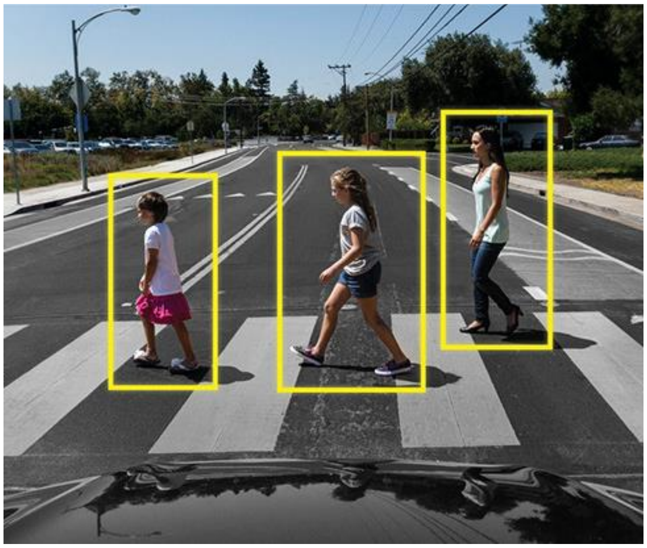


Image by NVIDIA via blogs.nvidia.com

Figure 1. Vision learning systems that misidentify objects could have catastrophic consequences

Systems and functions augmented with Artificial Intelligence/Machine Learning (AI/ML) promise to offer significant improvements over human operated functions. Assuring the successful and safe integration of AI/ML into everyday life is a challenge. Products being developed with AI/ML components have unique safety concerns for the system developer, product user, general public, and society at large. For example, an Advanced Driving System (ADS) or autonomous vehicle mishap on a public road could result in fatalities or injuries to occupants of the ADS vehicle, occupants of other vehicles, and pedestrians (Figure 1). While robust system and software safety processes exist that are used to identify and mitigate general risk, AI/ML components must be developed and integrated using a specialized and focused safety process. This specialized system safety process is required to characterize, analyze, and mitigate the unique aspects of AI/ML elements.

A-P-T Research, Inc. (APT) is a global leader in analyzing and applying system and software safety to Government and commercial programs. APT leverages its extensive background in system safety and software safety to identify specific areas where the specialized AI/ML domain challenges the traditional system and software safety paradigm. APT subject matter experts developed specific analyses to identify, assess, and mitigate hazard sources directly attributed to AI/ML components. These analyses are elements of an overall AI/ML system safety process that effectively targets safety deficiencies in a system or system of systems hosting AI/ML components.

This paper presents an overview of APT's AI/ML System Safety Process. The APT AI/ML System Safety Process focuses on detailed consideration of the design and operational domains defined for the AI/ML component and the data used to train and test the AI/ML model.

2.0 Background

System Safety is a specialized discipline within system engineering that is used to achieve an acceptable level of safety risk during all phases of the system's life cycle. Both government regulators and the system's program management set risk requirements and limits that must be met to assure public safety. System Safety professionals supporting the program will lead an integrated system safety program to assure the system meets specified safety risk levels.

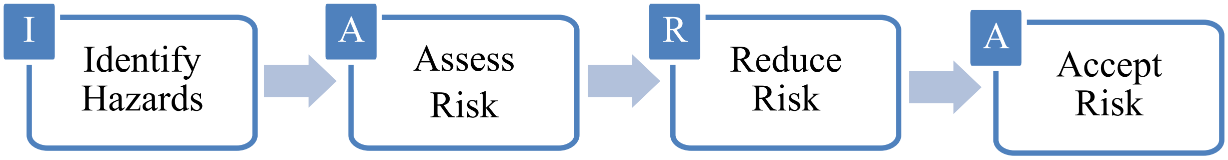


Figure 2. IARA: Iterative system safety process for mitigating risk

Figure 2 presents a general System Safety model that includes the four key System Safety elements: Identify Hazards, Assess Risk, Reduce Risk, and Accept Risk, or IARA.[1] APT developed and successfully applies the IARA model in government and commercial programs to achieve acceptable levels of safety.

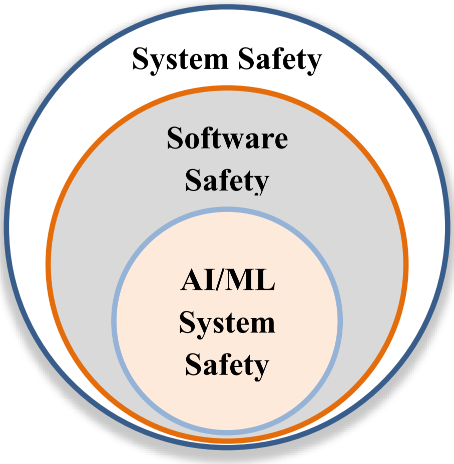


Figure 3. AI/ML system safety is a critical subset of a robust software safety program

If the system under study contains software that controls safety critical functions, it is common for a program to define software safety efforts in a Software System Safety Program Plan. Software safety is, therefore, a critical subset of a robust system safety program. Hazard sources and mitigations identified employing the IARA model standard as part of a software safety process are then fed back to the system safety program for disposition.

Although AI/ML components interface to the overall system and are embedded in a system's software, it is a mis-held belief that AI/ML components can simply be addressed by the software safety program. There are specific nuances AI/ML components present that software safety is not capable of capturing and must be addressed with a specialized system safety process. In addition, software engineers who are assigned to perform software safety may not be well versed in examining the validity of data used to train AI/ML models. Statisticians and data scientists must be experienced with analyzing key characteristics such as data domain and interfaces to the overall system. This necessitates a closer examination by safety engineers trained in data analytics of AI/ML components and assures any nondeterministic[2] outputs do not contribute to nor generate a hazard.

The Venn diagram shown in Figure 3 illustrates how AI/ML system safety fits into the overall system safety program architecture. It is important to note that while the last function of the

[1] APT teaches the fundamentals of the IARA process in its training course entitled "Risk Management for Safety Engineers"

[2] A nondeterministic algorithm is an algorithm that, even for the same input, can return different results on different runs. This makes validating acceptable results difficult since there is no single consistent answer.

IARA system safety model is to Accept Risk, the final action of the software safety and AI/ML system safety process is to verify and validate that mitigation measures reduce risk. Any residual risk after all mitigation techniques have been properly applied must flow up to the system safety process to be reassessed or accepted.

3.0 Unique Safety Characteristics within AI/ML

AI is a software element or model that, for a fixed set of inputs, can produce nondeterministic outputs, approximations, or multiple valid solutions. These outputs, produced by the AI component, are modeled predictions that the system will use to initiate an action or function. ML is an application of AI that entails the process of training the AI model with a known dataset of inputs and outputs to "learn," solve problems, and produce an output to narrowly defined problems with a limited scope. Therefore, when a fielded system is used in a real-world application, the trained AI model should make valid predictions within the operating domain using previously unseen inputs. Non-AI/ML software routines have definitive input and deterministic output with clear boundaries defined to certify the system within. The uncertainty in AI/ML component output makes current software safety processes ineffective in capturing hazard sources.

AI/ML components cannot use traditional safety risk assessment procedures that consider the deterministic severity and probability of a mishap to produce a risk rating. This is because a probability of a misclassification is challenging to determine with an AI/ML model and a misclassification by itself is a hazard source that may contribute to the severity of yet another hazard. New processes are required to quantify risk as a critical component of AI/ML system safety.

A unique characteristic of AI/ML that warrants specialized analysis is the dataset used to train and test the model. The data must be accurate, representative of the operating domain, and in sufficient quantity to adequately teach the AI/ML model. The dataset must have high fidelity, accuracy, and robustness within the operating domain. Datasets must also be examined for outliers that manifest themselves as data anomalies. The AI/ML component must also be examined for its ability to safely process these data anomalies. The quality of the dataset used directly corresponds to how well the AI/ML component performs.

In addition to examining the datasets for accuracy, the scope of the operating domain must be clearly established. It is imperative that the data used to train the AI/ML model covers the defined scope of the operating domain and must represent the operating domain established for the product/system functions. Any diversion between the operating domain and the dataset can cause significant anomalies in the AI/ML model results and therefore introduce safety risk.

4.0 APT's General AI/ML System Safety Process

The APT AI/ML System Safety Process is designed to provide specialized analyses necessary to identify, assess, and mitigate unique characteristics and risks associated with AI/ML implementation. Figure 5 presents the APT AI/ML System Safety Process.

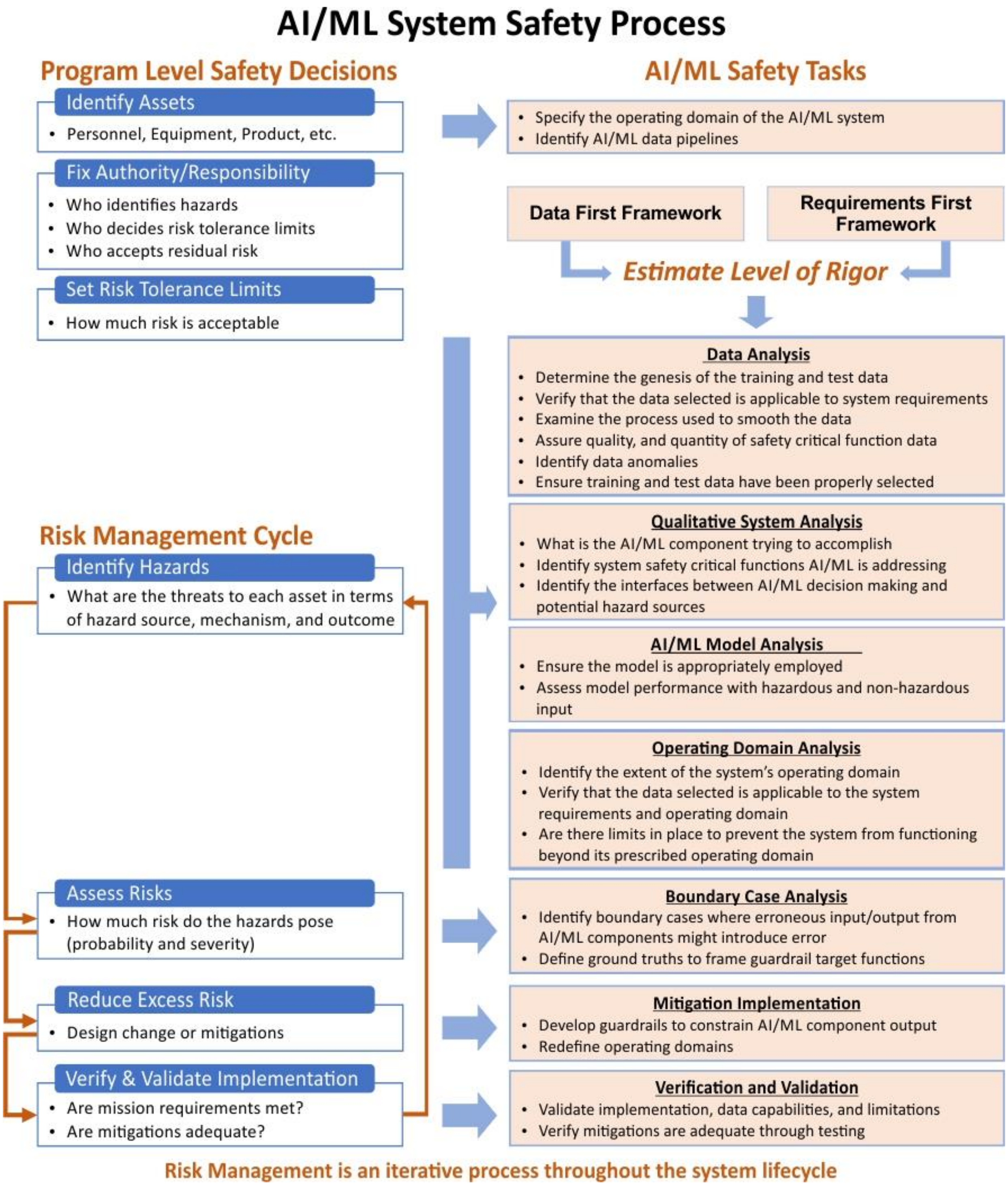
At the program level, it is imperative to determine the operating domain of the system, secure a robust operating domain data pipeline for training and test data, and ensure the proposed AI/ML model is sufficiently expressive. Focusing on identifying the operating domain and data source at the outset of the program scopes the AI/ML implementation and helps focus AI/ML model developers within the System Safety Program.

Image by Contractor Magazine via contractormag.com

Figure 4. AI/ML models with a data first framework are akin to finding a needle in a haystack

AI/ML components are typically developed based on either a ***Data First Framework*** or a ***Requirements First Framework***. In a Data First Framework, actionable decisions are made based on findings in a large set of pre-collected data (Figure 4). Data First AI/ML models are developed in the intelligence, business, logistics, and maintenance domains. In a Requirements First Framework, a requirement is identified before data are collected to train and test an AI/ML component. Requirements First AI/ML models are well suited for military weapons systems. While the AI/ML system safety process is not employed differently for either framework, it is instructive for safety personnel to know whether the genesis of an AI/ML component is data or requirements driven to help focus the subsequent analyses and risk management.

Starting with the baseline AI/ML component framework (Data First or Requirements First), a Level of Rigor (LoR) analysis is then performed to scope the AI/ML system safety process. LoR describes the depth and breadth of analyses and verification activities necessary to provide sufficient confidence that the AI/ML component and its safety-related functions will perform as required. Following an LoR determination, the risk management cycle can commence with an understanding of how detailed analyses must be to properly manage risk.



AI/ML System Safety Process

Program Level Safety Decisions

Identify Assets

- Personnel, Equipment, Product, etc.

Fix Authority/Responsibility

- Who identifies hazards
- Who decides risk tolerance limits
- Who accepts residual risk

Set Risk Tolerance Limits

- How much risk is acceptable

Risk Management Cycle

Identify Hazards

- What are the threats to each asset in terms of hazard source, mechanism, and outcome

Assess Risks

- How much risk do the hazards pose (probability and severity)

Reduce Excess Risk

- Design change or mitigations

Verify & Validate Implementation

- Are mission requirements met?
- Are mitigations adequate?

AI/ML Safety Tasks

- Specify the operating domain of the AI/ML system
- Identify AI/ML data pipelines

Data First Framework

Requirements First Framework

Estimate Level of Rigor

Data Analysis

- Determine the genesis of the training and test data
- Verify that the data selected is applicable to system requirements
- Examine the process used to smooth the data
- Assure quality, and quantity of safety critical function data
- Identify data anomalies
- Ensure training and test data have been properly selected

Qualitative System Analysis

- What is the AI/ML component trying to accomplish
- Identify system safety critical functions AI/ML is addressing
- Identify the interfaces between AI/ML decision making and potential hazard sources

AI/ML Model Analysis

- Ensure the model is appropriately employed
- Assess model performance with hazardous and non-hazardous input

Operating Domain Analysis

- Identify the extent of the system's operating domain
- Verify that the data selected is applicable to the system requirements and operating domain
- Are there limits in place to prevent the system from functioning beyond its prescribed operating domain

Boundary Case Analysis

- Identify boundary cases where erroneous input/output from AI/ML components might introduce error
- Define ground truths to frame guardrail target functions

Mitigation Implementation

- Develop guardrails to constrain AI/ML component output
- Redefine operating domains

Verification and Validation

- Validate implementation, data capabilities, and limitations
- Verify mitigations are adequate through testing

Risk Management is an iterative process throughout the system lifecycle

Figure 5. AI/ML specific tasking performed as part of a risk management process

4.1 IDENTIFY HAZARDS

The first step in the risk management cycle is to identify hazard sources posed by the AI/ML component. These hazard sources, if not mitigated at the AI/ML component level, will be missed at the software safety level and may induce additional hazard mechanisms and outcomes. To help identify AI/ML component hazard sources, APT employs four main analyses: Data Analysis, Qualitative System Analysis, AI/ML Model Analysis, and Operating Domain Analysis. These analyses target the four critical components of AI/ML component development. Any deficiencies in an AI/ML model, dataset used, or error in defining an operating domain will introduce hazard sources.

Data Analysis – This analysis determines the genesis, quality, and quantity of the training and test data and verifies that the data, datasets, and pipelines selected are applicable to system requirements. Examining the process used to smooth the data will identify issues with dimensionality, causation, and data mining. Finally, there is potential to identify data anomalies that may have introduced errors in AI/ML model training that result in hazards.

Qualitative System Analysis – This analysis examines the overall integration of an AI/ML component into its parent system. The analysis identifies what the AI/ML component is designed to accomplish, how results from the AI/ML component influences system safety critical functions, and identifies interfaces between AI/ML component decision making and potential hazard sources.

AI/ML Model Analysis - This examination of the AI/ML model will provide insight as to the robustness of the results. A properly designed model will neither have too few layers and nodes (underfitting) nor have too many layers and nodes (overfitting). The AI/ML model will also be probed to see how it handles both hazardous and non-hazardous input.

Operating Domain Analysis – This critical analysis identifies the extent of the system’s operating domain. Significant hazard sources may be introduced by an AI/ML component that is called to operate outside the operating domain for which it has been trained. This analysis provides verification that the training and test data selected are applicable to the system requirements and operating domain. Ultimately, it is the goal of this analysis to verify that guardrail limits are in place to prevent the system from functioning beyond its prescribed operating domain.

Hazard sources identified and collected as part of these four analyses are documented and tracked as part of the software safety or program level system safety efforts. The benefit is that the entire system safety program is now capable of identifying and mitigating the hazard source that leads to a safety critical mechanism and outcome from all sources within the system. The depth and breadth of each analysis is determined by the LoR analysis and will contribute to the cost and manpower requirements assessment necessary to complete the AI/ML system safety process.

4.2 ASSESS RISK

It is well understood in software safety that software code error, by itself, is not considered a hazard. Rather, the software will return a result or command that instantiates a hazard mechanism and ultimately a hazardous outcome. To complicate matters, it is difficult to quantitatively assess risk due to software errors since the probability of occurrence is hard to quantify.

AI/ML components are even more troublesome in that their results are nondeterministic. The key is to understand the range of output and determine if all output within that range is properly bounded. If results from an AI/ML component are not properly examined downstream, there is potential for considerable risk.

Boundary Analysis – This analysis identifies interface and dataset cases where erroneous inputs or outputs to and from an AI/ML component could introduce entire system-level or subsystem-level errors. The analysis will identify those interface errors that could cause a system/sub-system mishap. A boundary analysis will also define ground truths to frame guardrail target functions for future risk reduction options.

In some instances, the program will designate a desired success rate of the AI/ML component. Results of this testing can sometimes be factored into the risk assessment as a probability of occurrence; however, it is important to note that even a slight deviation in operating domain can cause significant deviations in the success rate of an AI/ML component.

It is imperative that software safety engineers collaborate with AI/ML safety engineers when performing a boundary analysis. AI/ML safety engineers can provide useful data on what the potential mis-characterization rate of an AI/ML component may be, and software safety engineers can determine if the results of an AI/ML component will contribute to a safety critical hazard. Hazard sources generated by the AI/ML component that contribute to significant risk are identified at this stage. Those hazards are forwarded to the risk reduction phase of the risk management cycle.

4.3 REDUCE EXCESS RISK

Risk reduction solutions for AI/ML components will generally follow two approaches, dataset updates and guardrails. Data updates will correct issues identified with training datasets that may not adequately represent the operating domain or include all expected hazards. Guardrails are usually software controls that prevent the AI/ML component from providing an output that is outside specified parameters or deviates from an established ground truth. A risk reduction solution may follow several approaches to either eliminate or mitigate an AI/ML safety risk.

4.4 VERIFY AND VALIDATE IMPLEMENTATION

The last phase of the risk management cycle is to verify and validate that the risk reduction techniques and implementation are applied properly. AI/ML system safety engineers will validate mitigation implementation, data capabilities, and limitations of the AI/ML component and then verify that the mitigations applied are adequate through testing.

Once verification and validation have been completed, the AI/ML system safety engineers will circle back to the hazard identification phase of the risk management cycle and ensure the risk posed by the AI/ML component is acceptable.

5.0 Benefits of APT's AI/ML System Safety Process

The APT AI/ML system safety process provides a robust, systematic approach to manage system safety risks for an AI/ML based product that dovetails to APT's overall System Safety Program. The AI/ML system safety process can be tailored for product-specific applications and can be implemented to manage AI/ML safety risks at the beginning of product development, later in product development effort, and after product deployment to encompass the entire system lifecycle. APT brings significant experience and expertise to bridge the AI/ML safety domain into the system safety and software system safety domains, ensuring a robust yet cost-effective AI/ML system safety process to identify, assess, reduce, and accept system risk.

6.0 About APT

A-P-T Research, Inc. (Analysis, Planning, Test Research, or "APT") is a 100% employee-owned, small business based in Huntsville, AL. APT provides professional engineering and programmatic services in a variety of disciplines including systems engineering, risk assessment and analysis, test planning, range safety, system safety, software system safety, explosives safety, industrial and quality engineering, quality assurance, mission assurance, Independent Verification and Validation, software development and modeling, and related areas. APT currently supports more than 50 customers of which approximately 40 are Government agencies. Our personnel are owners and each shares in the commitment to APT's corporate mission of continually providing state-of-the-art expertise and ensuring the highest level of customer satisfaction.

APT has a 20-year track record as a demonstrated national leader of assessing software quality, evaluating software safety, and conducting independent and integrated software verification and validation. The results of these efforts have served to ensure Government and critical commercial technologies have been successful for medical, flight, defense, and other safety critical applications. Our software safety personnel use an APT-developed and proven process to identify safety critical elements and audit software development supporting qualification and acceptance. APT has applied its expertise and experience in software safety to develop a focused system safety process for AI/ML applications. For more details on how APT can help your program with nondeterministic components, contact John Hall, Ph.D., Technical Director, (256) 327-3379, jhall@apt-research.com.

CONTRIBUTING AUTHORS

Thomas Neild. Mr. Nield is a consultant as well as a writer, conference speaker, and trainer. After a decade in the airline industry, he authored two books for O'Reilly Media and regularly teaches classes on artificial intelligence, statistics, machine learning, and optimization algorithms. Currently, he is assisting APT Research while teaching at University of Southern California, defining system safety approaches for artificial intelligence. He enjoys making technical content relatable and relevant to those unfamiliar or intimidated by it.

John Hall. Dr. Hall has over 38 years of experience in the Army and in the commercial aerospace industry. As an Army officer, Dr. Hall's assignments included posting as an Operations Research System Analyst where he supported operational testing and supply chain management efforts. He has been with APT for over 20 years and has provided support to Government and commercial customers in the areas of system safety, industrial engineering, and quality management.

Wayne Devoid. Mr. Devoid has 20 years of experience in flight safety for suborbital and orbital rocket launches. He pioneered APT's quantitative risk analysis methodology and software tools for unmanned systems. Mr. Devoid is excited to apply his system safety background for unmanned and manned systems to the AI/ML domain.

Tim Middendorf. Mr. Middendorf has over 35 years of experience between his career as an Air Force Officer as well as in the commercial industry. He has worked many facets of the aerospace industry including space systems engineering, system safety, and space operations, to include Range Flight Safety, UAV flight analysis, propulsion technology system analysis, spacecraft engineering/operations, and spacecraft launch operations. Over 20 years of his experience centered on supporting the FAA Office of Commercial Space Transportation (AST). He directly supported the NASA Headquarters Office of Range Flight Safety while working at Kennedy Space Center.

Jeff Richards. Mr. Richards is Director of the APT Software Engineering Division. He has over 20 years of software and safety experience, working a variety of programs from manned and unmanned aviation, ground vehicles, underwater, and rocket and missile systems.

Clark Kilgore. Mr. Kilgore has been with APT for over 20 years supporting various missile systems within the Missile Defense Agency. Prior to joining APT, Clark worked in the commercial field as well as NASA. He has 10 additional years of experience in Systems Engineering, manufacturing, and packaging.