Trustworthiness for AI in Defence (TAID)

-

Reference Frameworks and Toolkits Examples Annex

Submission date: 09/05/2025

Document version: 1.0

Authors: TAID Working Group (TAID WG)

Table of Contents

Table of Figures

1. Frameworks

1.1 Ten-step process for establishing the trustworthiness of AI-based Systems.

The increasing use of artificial intelligence (AI) and machine learning (ML) technologies in software, hardware, data, and human defense systems introduces fundamental vulnerabilities and risks due to dynamic and unreliable behavior. AI/ML systems learn from data, bringing known and unknown challenges to the behavior of these systems and their interaction with their environment. Currently, AI/ML technologies are being developed in isolation, which presents a significant risk. Models and algorithms are developed in test environments that are isolated from real-world environments and without the context of larger systems or broader products into which they will be integrated for deployment. A significant issue is that models are typically trained and tested on only a few curated datasets, without measures and safeguards for future scenarios, and without consideration of downstream tasks and users.

To move away from current approaches and procedures, a dedicated process is required to reduce the existing risks. To achieve trustworthiness, a 10-step process is required to develop an AI/ML-based military defense system and to perform the necessary verification and validation.

At the conclusion of each phase, a dedicated review and testing period is conducted. This period serves to present the technical developments in conjunction with the requirements and the corresponding verification and validation steps. Additionally, it allows for the critical decision-making process regarding the direction of future development, as well as the schedule. Finally, a debriefing of the process is conducted.

In each phase, special verification and validation (V&V) measures and tests are specified at the respective level.

1.1.1 Level 0 First Principles

This level of AI research is initiated by a novel idea, a pivotal question, or a problem viewed from a novel perspective in relation to the specified requirements. The work primarily involves extensive literature research, the development of mathematical foundations, mind-mapping concepts and algorithms, and the acquisition of an understanding of the data—for work in theoretical AI and ML.

It is essential to create considerations and concepts for the required data (sources, references). Uncharted territory can present greater challenges than well-trodden research paths, as is the case in computer vision with neural networks. It is therefore beneficial to have a frame of reference when embarking on a new path in AI/ML research.

Level 0 Data: Not necessary.

Level 0 Review/Test: The initial review and examination of a given topic is conducted by the lead reviewer, who is typically the head of the research laboratory or team. Hypotheses and investigations are evaluated for mathematical validity and potential novelty or usefulness with respect to military requirements and operational capabilities. This evaluation does not necessarily concern the code or results of experiments. The military organization is eligible to participate in the review.

1.1.1 Level 1 Goal orientated Research

To transition from the fundamental aspects of a model or algorithm to practical applications, low-level experiments should be designed and conducted with the specific properties of the model or algorithm in mind (as opposed to end-to-end runs for a performance benchmark). This includes the collection and processing of sample data to train and evaluate the model. The sample data does not necessarily have to be the complete data; it may be a smaller sample that is currently available or easier to collect. In some cases, it may be sufficient to use synthetic data as a representative sample. The analysis of sample data can provide a framework for the collection and processing of data (including the assessment of the feasibility of collecting all necessary data).

Initial approaches should be developed for the creation of scenarios that can be used for practical demonstration of the results obtained at this level. These scenarios should be designed in a way that allows for reuse in refinements and extensions in subsequent levels.

The experiments, regardless of their outcome, and the mathematical foundations must undergo a review process with other customer researchers and military professionals before advancing to Level 2. The objective is to conduct preliminary experiments in a timely and expedient manner, with the intention of identifying potential areas for improvement. Therefore, so-called "dirty code" is acceptable at this stage, and in fact, full test coverage is discouraged until the entire code base is organized and maintainable. It is of paramount importance to commence semantic versioning at the earliest possible stage of the project lifecycle, which must encompass code, models, and data sets. This is of critical significance for retrospectives and reproducibility, as issues with this in later phases can be costly and severe. It is of the utmost importance to record this versioning information and further progress on the TRL card. It is of the utmost importance to consider IT security and safety in operational use at an early stage and to provide appropriate mechanisms.

It should be noted that hidden feedback loops are a common and problematic phenomenon in real-world systems that influence their own training data. Over time, user behavior may evolve to select data inputs that they prefer for the AI system, which represents some deviation from the training data.

Estimating uncertainty is valuable in many AI scenarios, but it is not easy to implement in practice. This becomes even more complicated when there are multiple data sources and users, each of which introduces a generally unknown amount of noise and uncertainty.

The components of an MLOps platform may vary, but they typically include managed notebooks for collaborative development, feature and vector pipelines for efficient data handling, and feature stores and vector stores for organizing and retrieving data features. Model registries track and manage different versions of models, while model training and serving infrastructure handles the computation-heavy tasks of training models and making them available for predictions. Finally, model monitoring tools continuously assess the performance of deployed models to ensure their accuracy and reliability over time.

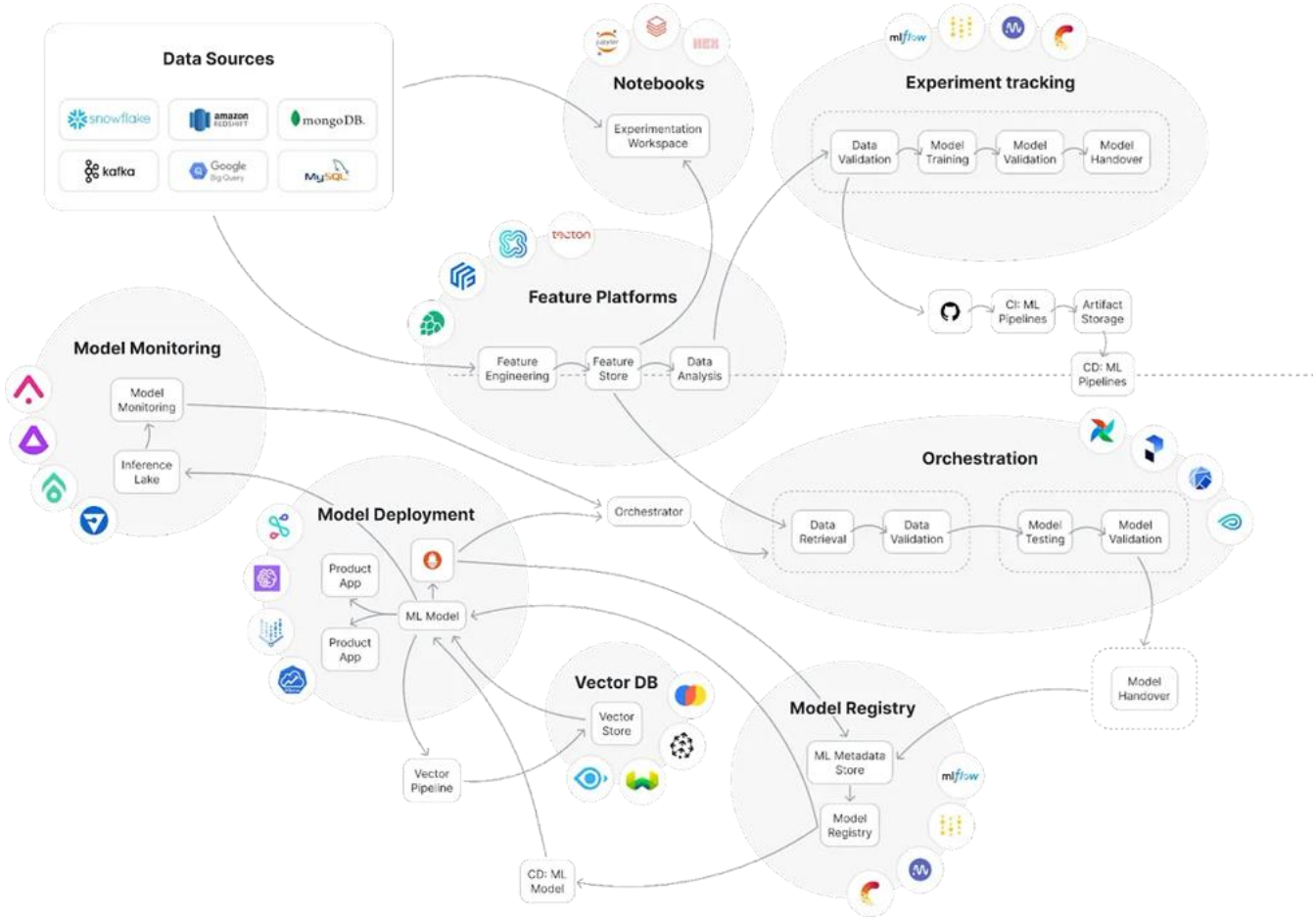

Figure 1 - Typical components of an MLOps platform

The process of invention begins with the observation of basic principles. Once these principles have been identified, practical applications can be developed. These applications are based on assumptions that may or may not be proven or supported by detailed analysis. The examples provided are limited to analytic studies.

Level 1 data:

To ensure accuracy, it is essential to work with representative sample data that reflects the downstream real datasets. This can include a subset of the actual data, synthetic data, or a combination of both. It is important to define data collection and processing strategies early on to avoid obstacles in the future. Additionally, when conducting low-level AI/ML experiments, it is crucial to maintain objectivity and avoid biased language.

Level 1 Review/Test:

The review panel is comprised solely of members of the research team and technical military reviewers (customer). Their role is to evaluate the scientific rigor of the initial experiments and highlight important concepts and previous work from their respective fields. Feedback may be provided in multiple iterations, and additional experiments may be required.

1.1.2 Level 2 Proof of Principle

During this phase, active research and development is conducted. This is primarily achieved through the development and operation of test environments. These may be simulated environments and/or simulated data that closely resemble or are identical to the conditions and data of real scenarios (e.g., when using historical data). It should be noted that the scenarios are primarily driven by model-specific technical objectives and not necessarily (yet) by application or product objectives. A significant output of this phase is a formal document outlining the research requirements, which must be accompanied by well-specified V&V steps. The research requirements must be contextualized within the broader military requirements.

One approach to validate the earlier theoretical developments was to generate synthetic data to isolate specific features in the data that are expected to be represented in the latent manifold. The results may prove promising for anomaly detection. For instance, the latent representation of the data could be employed to automatically identify hit images that deviate from the norm (anomalous), while the manifold could be utilized to investigate the semantic differences between them. Based on an implicit probabilistic modeling approach, uncertainty estimation could be a valuable downstream function. This represents one of several pivotal junctures in the broader process. The R&D team considers a multitude of potential avenues and charts a course of action.

a) Development of a prototype for Level 3.
b) Continuation of R&D for longer-term research initiatives. Alternatively, a combination of a and b may be considered.

Regarding the scenarios to be used, it is necessary to make concrete considerations and specifications regarding the clustering of the infinite test space of an AI/ML application in a technically meaningful way. It is of the utmost importance to consider the implications of such applications in everyday life, particularly in critical areas. A comprehensive risk assessment must be conducted regarding scenarios that have not been considered or performed.

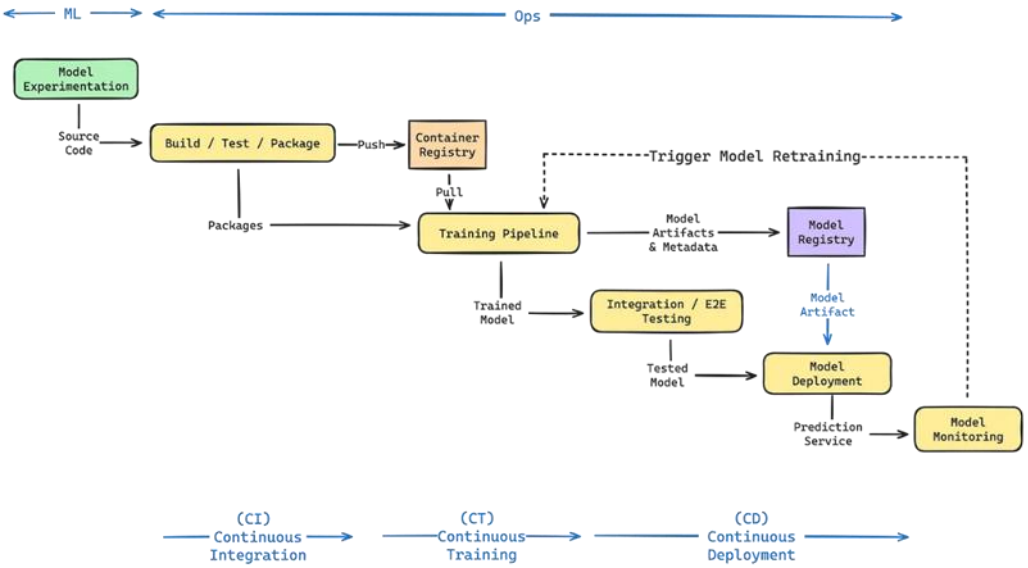

Figure 2 - Continuous Deployment

In the context of machine learning (ML) applications, the continuous integration (CI) process integrates numerous principles of traditional software CI, while also exhibiting a distinct focus and scope. The primary objective of ML CI is to ensure code quality and functionality, thereby preparing the codebase for subsequent phases of the ML lifecycle. The separation of model training and evaluation from the CI pipeline in ML is driven by several key reasons, including complexity and resource requirements. Model training often necessitates the use of significant computational resources, such as specialized hardware like GPUs. Integrating such resource-intensive tasks into the CI phase is impractical and could impede the efficiency of the code integration process. By decoupling the model training phase from the integration process, there is greater flexibility in the development workflow. Training can be conducted independently with various parameters and evaluation methods. Simultaneously, CI can proceed without hindrance, allowing for direct deployment of pre-existing trained models without the need for retraining during each integration cycle. The development of machine learning (ML) models is inherently iterative, often involving experimentation with different parameters and methods. The inclusion of this iterative training in the continuous integration (CI) workflow could considerably impede the rapid iteration and integration that CI aims to achieve, thus slowing down the process. Therefore, the outcome of the CI phase in ML is a packaged model code that is prepared and ready for deployment in either a prediction serving or a training environment. This separation ensures that the model is prepared for training, evaluation, and eventual deployment, following the specific requirements and workflows inherent in ML development.

Continuous Training (CT) is a pivotal component in the ML lifecycle, focusing on the ongoing process of training and retraining ML models. This process is essential in keeping models relevant and effective in the face of evolving data and changing environments.

Continuous Deployment (CD) in ML is the process of automatically deploying ML models to production after they are trained and validated. This process ensures that the latest, most effective version of the model is always in use, thereby improving the overall efficiency and performance of the system.

The process of active research and development has commenced. This involves the conduct of analytical studies and laboratory studies with the objective of physically validating the analytical predictions associated with discrete elements of the technology. Illustrative examples include components that have not yet been integrated or that represent a non-representative sample.

Level 2 data:

In this phase, datasets may include publicly available benchmark datasets, semi-simulated data based on the Level 1 data sample, or fully simulated data based on specific assumptions about potential operational environments. The data should allow researchers to characterize the properties of the model and highlight boundary cases or constraints to illustrate the military utility of further research and development of the model.

Level 2 Review/Test:

To progress to Level 3, the technology must be capable of supporting the research claims made in the previous stages. This must be demonstrated both quantitatively and qualitatively by the proof-of-principle data. Furthermore, the technology must be capable of undergoing well-documented and reproducible analyses.

1.1.3 Level 3 Systems Development

During this phase, there are checkpoints that guide code development towards interoperability, reliability, maintainability, security, extensibility, and scalability. The code should be prototypical, representing a significant improvement over research code in terms of robustness and cleanliness. It should be well-designed with a good architecture for data flow and interfaces, generally covered by unit and integration tests, state-of-the-art, and well-documented. It is possible that programmers may still assume that this code will be reworked or scrapped for production. The prototype code is relatively primitive in terms of the efficiency and reliability of the eventual system. As the working group transitions to Level 4 and proves the concept, it is important that product development is involved in defining the service level agreements and objectives (SLAs and SLOs) of the eventual production system.

Proof-of-concept development and testing can demonstrate promising opportunities for several defense-relevant military applications. This underscores the need for several key capabilities, which are defined as research and development (R&D) and product requirements. These capabilities include interpretability, which is necessary for end-user confidence; uncertainty quantification, which is necessary to represent confidence levels; and human-in-the-loop, which is necessary for expertise. To prevent the deferral of capabilities to beta or acceptance testing, or the complete omission of them, it is essential to incorporate AI/MLTRL PoC steps and review processes. The definition of normal use cases and the specification of critical applications in their derivation are crucial for the clustering of the infinite test space by scenarios.

The core steps of the criticality analysis involve a three-step process, as illustrated in the figure below. First, relevant influencing factors, known as criticality phenomena, are extracted. Second, the understanding of critical phenomena is enhanced by identifying underlying causal relationships. Finally, abstraction and classification of causal relationships are utilized for scenario space condensation. The objective when considering a critical phenomenon is to improve understanding of the underlying causal relations. To achieve this, a plausible causal model is developed to explain how this phenomenon heightens criticality. An expert first proposes a hypothetical causal relationship to account for the phenomenon. Empirical analyses are then used to collect evidence supporting the plausibility of the proposed causal relationship. The hypothesis is refined through iterative learning, which involves expanding the dataset, continuously updating the ontology, and utilizing metrics and simulation models. If the statistical evidence is sufficient to verify and validate a causal relationship, it is considered a credible explanation for the phenomenon. To analyze maritime transport segregation based on criticality, it is essential to first identify the critical points. The utilization of available knowledge or data can lead to a more confident and accurate analysis.

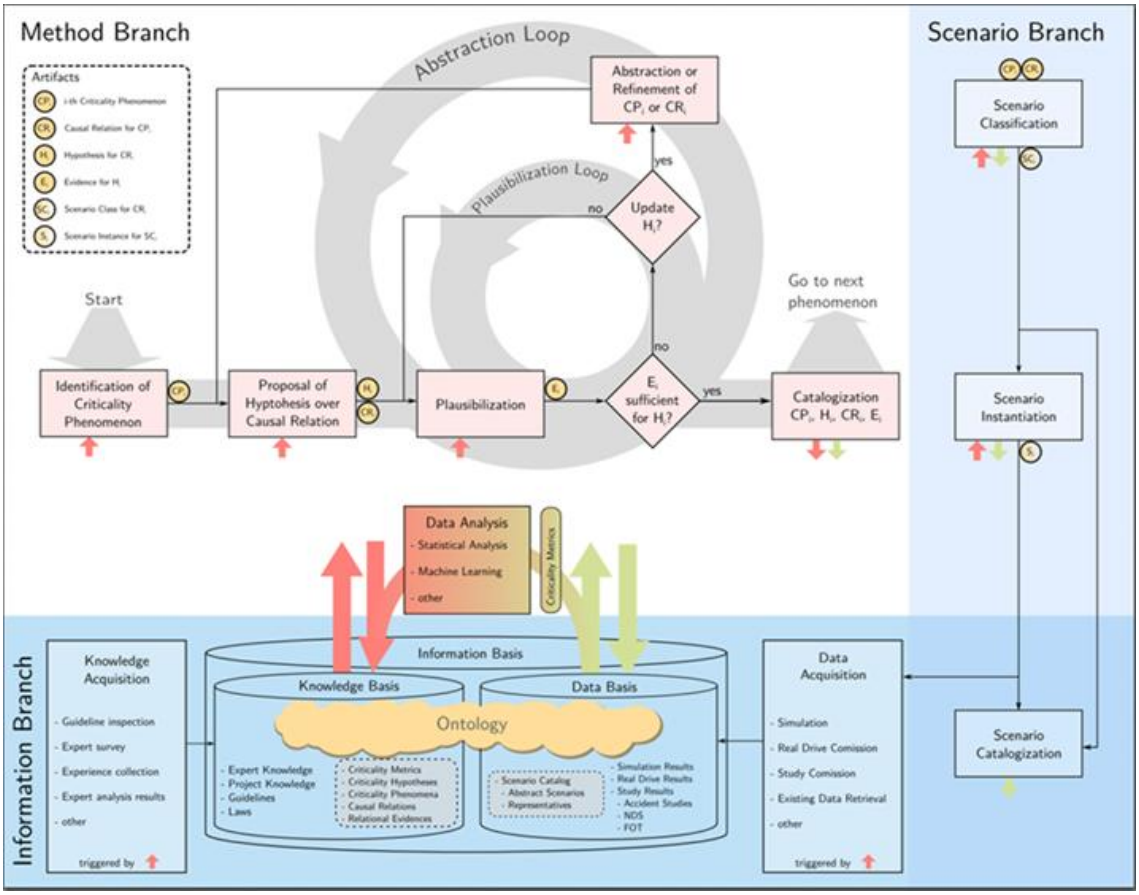

Figure 3 - Core Steps of Criticality Analysis

The integration of basic technological components is employed to establish their functional interdependence. This approach is relatively "low fidelity" in comparison to the eventual system. Illustrative examples include the integration of ad hoc hardware in the laboratory.

Level 3 data:

This is approximately equivalent to Level 2. In general, the review of the previous phase can identify potential gaps in data coverage and robustness that need to be addressed in the subsequent phase. However, for the test suites developed in this phase, it is useful to define certain subsets of the experiment data as standard test sources and to set up mock data for specific features and scenarios to be tested.

Level 3 Review/Test:

The review will be conducted by experts in the fields of applied AI and engineering, with a focus on sound software practices, interfaces, and documentation for further development, as well as version control for models and datasets. Any domain- or organization-specific considerations for future data management will be highlighted in this review.

1.1.4 Level 4 Proof of Concept

In this phase, the focus is on the development of applications that are oriented towards practical applications. The objective is to demonstrate the technology in real-world scenarios. Rapid

proof-of-concept examples should be developed to explore potential application areas and to communicate the quantitative and qualitative results. It is important to use real and representative data for these potential applications. For the proof of concept, data engineering involves scaling up data collection and processing from Level 1. This includes the collection of new data or the processing of all available data using scaled experimentation pipelines from Level 3. In some scenarios, new data sets are used for the PoC, such as those from an external research partner or the military for validation. To transition from sample to real data, it is important that the experimental metrics also evolve from AI/ML research to the applied environment. Proof-of-concept evaluations should assess model and algorithm performance, including precision and recall for different data splits, as well as computational cost, such as CPU vs. GPU runtimes. Additionally, it is important to consider metrics that are more relevant to the end user, such as the number of false positives in the top N predictions of a recommender system. At this stage, the PoC study should highlight the distinctions between clean and controlled research data and noisy and stochastic real-world data, as well as the limitations of utilizing such data. It is crucial to elucidate the rationale behind the selected scenarios, their relative significance, and the manner in which the expansive test space was stratified.

The fidelity of the breadboard technology markedly improves. The fundamental technological components are integrated with relatively realistic supporting elements, thus enabling their testing in a simulated environment. Examples of this include the high-fidelity laboratory integration of components.

Level 4 Data:

In contrast to the preceding phases, it is of paramount importance that the PoC be based on genuine and representative data. Even with methods to verify that the data distributions in the synthetic data reliably reflect those in the real data, it is necessary to achieve sufficient confidence in the technology with real data from different use cases. Furthermore, it is essential to consider how to obtain high-quality and consistent data required for future model inference. A proof of concept (PoC) data pipeline should be created to resemble the future inference pipeline. The pipeline should accept data from the intended sources, transform it into features, and send it to the model for inference.

Level 4 Review/Test:

A demonstration of the benefits for one or more real-world applications (each with multiple datasets) must be performed, the assumptions and limitations defined, and the data maturity reassessed. This includes evaluating the real data for quality, validity, and availability. During the test/review, security and privacy considerations are also evaluated (if necessary).

These aspects should already have been defined in the requirements document with a risk quantification. If not, a separate document should be prepared. This document should be seen as a useful mechanism for mitigating potential problems.

1.1.5 Level 5 AI Capability

At this stage, the technology is more than an isolated model or algorithm; it is a specific capability. An interdisciplinary working group may be formed because the development of technology must begin in the context of a larger real-world process. That is, the model or algorithm moves from being an isolated solution to being a module of a larger application.

Reaching Level 5 can be quite difficult because it means increasing the resources needed to bring the AI/ML technology to product maturity. This includes the consideration of extensive security measures in the data-driven global architecture. A common approach in this stage is A/B testing, where decisions made by the new model are compared against those made by the current model. This comparison helps in evaluating whether the new model marks an improvement or shows regression. In the MLOps pipeline, models are first deployed to a staging or shadow environment. This environment is a replica of the production setup, designed to mimic real-world conditions as closely as possible. During a shadow deployment, the new model runs in parallel with the existing production model. However, it does not affect actual operational decisions or outputs. This setup enables an observational evaluation of the model in realistic conditions, without risking current operations. The main advantage is the ability to validate the model in a safe and controlled environment, which provides valuable insights into its performance in production.

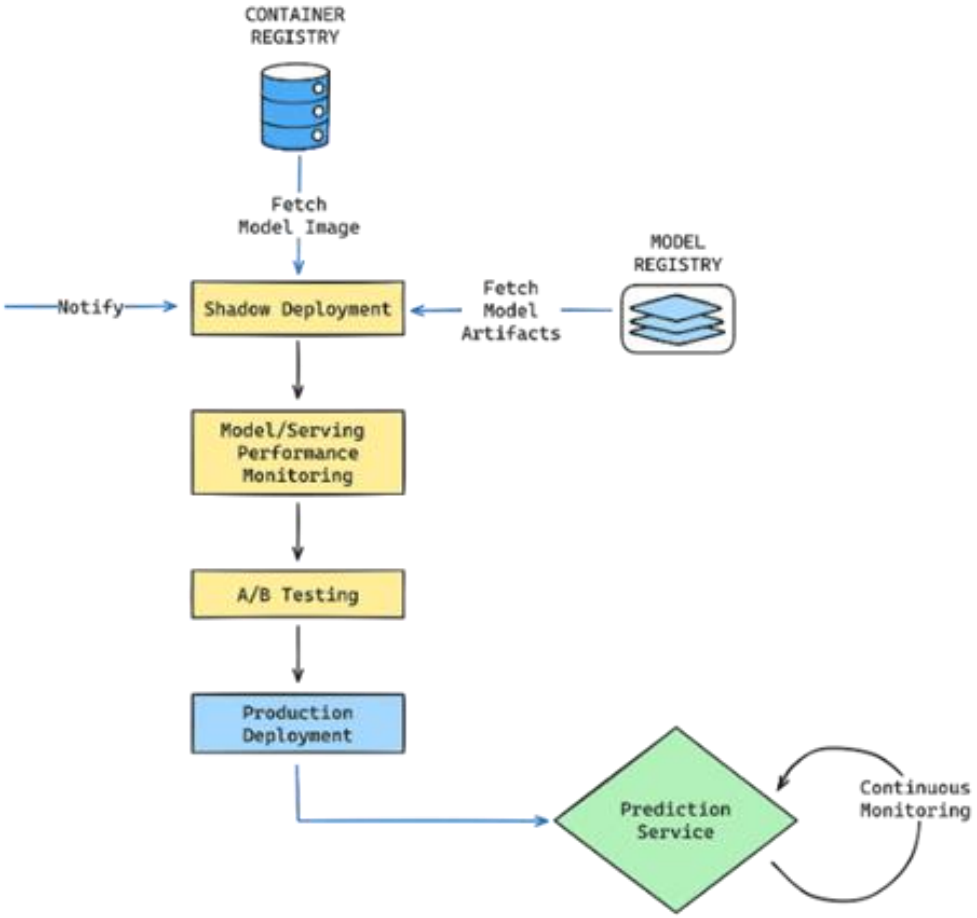

Figure 4 - Continuous Deployment

Once deployed in this preliminary environment, the model's performance is closely monitored. This involves tracking key metrics critical for the model's success, such as accuracy, latency, throughput, and specific business KPIs. Monitoring in this phase is crucial to ensure that the model behaves as expected and meets the set standards before it impacts real users.

Level 5 Data:

This data set is mostly consistent with level 4 characteristics. However, it is important to consider the scaling of data pipelines due to the increased usage by military users, including automated testing in later phases. However, it is important to consider the scaling of data pipelines due to the increased usage by military users, including automated testing in later phases. However, it is important to consider the scaling of data pipelines due to the increased usage by military users, including automated testing in later phases. Scaling can create data management challenges. It is also important to note that data pipelines may not necessarily reflect the structure of the teams/users or the overall military applications. This document defines the challenges that arise from data silos, overlap, unclear responsibilities, and a lack of control over the entire data lifecycle. It outlines different approaches to data governance, including planning and control, organizational, and risk-based strategies.

Level 5 Review/Test:

In this phase, it is of the utmost importance to complete all verification and validation (V&V) activities and steps that were initiated in earlier R&D phases, especially in Phase 2. Furthermore, the product-related requirements and corresponding V&V must be finalized. Thorough testing is conducted to ensure military acceptance at the earliest possible stage of product realization, well before launch. It is of significant consequence to document the division of the infinite test space into scenarios. The rationale behind the selection of critical and borderline scenarios must be clearly articulated, along with a justification for their inclusion. The normal scenarios should align with the intended application area of the AI/ML system, as defined by the requirements. Furthermore, a comprehensive and transparent analysis of the potential risks associated with scenarios that are not being tested must be provided.

1.1.6 Level 6 Application

At this level, the primary task is software development to elevate the code to a product level. The resulting code will be accessible to users upon acceptance. Therefore, it must adhere to precise specifications, have comprehensive test coverage, and well-defined APIs. Additionally, the resulting AI/ML modules must be made robust for multiple target use cases, depending on the requirements. If these target use cases require explanations of the model, the methods must be developed and validated along with the AI/ML model. Moreover, they should be tested for their effectiveness in faithfully interpreting the model's decisions, particularly in the context of downstream tasks and end users. This is important because there is often a discrepancy between AI/ML explanations for AI/ML specialists and actual military users. Similarly, when designing AI/ML modules, it is crucial to consider known data challenges, especially in testing the model's robustness and the broader pipeline's ability to handle changes in data distribution between development and deployment.

The product requirements document must provide a comprehensive account of the deployment environment(s), as the term "AI/ML deployment" (or "deployment") is often used in a manner that is ambiguous and requires careful consideration. There are two main types of deployment: internal, which refers to the use of APIs for experimentation and other purposes primarily by data science and AI/ML teams, and external, which refers to the deployment of AI/ML models to be embedded in or used by a real application with real users. When comparing cloud, on-premises, or hybrid deployments, batch and streaming solutions, open-source solutions, and containerized applications, the conditions under which they are deployed can vary significantly. Additionally, data may be restricted for compliance reasons during deployment, or access may

only be available to encrypted data sources, some of which may only be accessible locally. These scenarios can hinder advanced AI/ML approaches, such as federated learning, and other privacy-focused AI/ML applications. Depending on the application, an AI/ML model may not be able to operate across borders. This implies that the AI/ML model is embedded in a rule engine workflow, acting as an advisor to discover edge cases in the rules. Operational factors, which are not typically considered in model and algorithm development, can be important in military applications and have a significant impact on modeling and algorithm selection.

It is therefore of critical importance to make system decisions in Phase 6. It is advisable that these decisions are not made too early in the deployment process, when the specific scenarios and requirements are not yet fully defined. Similarly, it is inadvisable to make these decisions too late, as this could result in delays or failures in deployment.

Level 6 Data:

During this phase, it is crucial to gather and operationalize additional data to enhance the robustness of AI/ML models, algorithms, and their surrounding components. This entails collecting negative examples in the form of negative scenarios to test local robustness, identify semantically equivalent errors, and eliminate them. These measures validate the model's consistency with domain assumptions and generalize data collection from various sources to assess the trained model's response. In domains such as military applications, where data access is limited, these considerations and validation measures are of even greater importance.

Level 6 Review/Test:

The objective of this phase is to verify the quality of the code, the newly defined technical product requirements, the Service Level Agreement (SLA), and Service Level Objective (SLO) requirements for the system, the specification of the data pipelines, and, if necessary, revise the AI ethics. It is important to approximate real use cases. It is crucial to comply with data privacy and security laws, as missteps in compliance can result in significant consequences, including project failure.

The following core elements define a healthy model monitoring framework for models:

1. Performance Metrics Tracking: Continuous measurement of key metrics such as accuracy, precision, and recall, guaranteeing the model is performing as expected.
2. Monitoring for data drift (changes in input data) and model drift (changes in model performance over time), both of which can signal a need for model updates or retraining.
3. Anomaly Detection: The identification of unusual patterns or inconsistencies in the model's outputs or input data may indicate the presence of potential issues.

1.1.7 Level 7 Integration

To integrate the developed technology into existing military systems, it is recommended that a balanced working group be formed, consisting of infrastructure engineers and applied AI engineers. This phase of development is highly susceptible to latent model assumptions and failure modes, and therefore cannot be developed by software engineers alone. It is therefore crucial that joint development of important tools and tests be carried out.

Tests that run application-specific critical scenarios and data sections are required. To quantify the risks involved, an appropriate risk quantification table must be created.

- A "golden data set" should be defined to validate the performance of each model and model sequence for use in continuous integration and deployment testing. Metamorphic testing, a software engineering method for testing a specific set of relationships between the outputs of multiple inputs, should be performed when integrating AI/ML modules into larger systems.
- A codified list of metamorphic relationships can provide valuable verification and validation measures and steps when integrating AI/ML modules into larger systems.
- Data intervention tests are employed to identify data errors at various points in the pipeline. These tests are conducted both downstream to assess the potential impact of data processing and AI/ML on consumers or users of that data and upstream during data input or creation. Rather than using model performance as an indicator of data quality, it is important to employ intervention tests that detect data errors with specific data validation mechanisms.

These tests are useful in reducing under-specification in AI/ML pipelines, which is a significant obstacle to reliably training models that behave as expected in practice. Quality assurance (QA) engineers play a crucial role in ensuring reliability, monitoring data processes for privacy and security at Level 9, and performing audits for downstream accountability of AI methods.

Level 7 Data:

In addition to the test suite data discussed above, this level requires the QA team to prioritize data governance. This encompasses the acquisition, management, utilization, and protection of data by the organization. The significance of this was previously alluded to in Level 5 to circumvent associated technical debt, and it is of paramount importance at this pivotal integration nexus. This may present additional governance challenges due to the downstream effects and consumers.

Level 7 Review/Test:

The review or test should concentrate on validating the data pipelines and test suites. A scorecard, such as the ML Testing Rubric [4], can be useful in this regard. Ethical considerations should also be addressed at this level, if necessary, as it is easier to address them now, when many test suits are already in place, than later, just before delivery or acceptance.

1.1.8 Level 8 System Ready

It is essential that the technology be demonstrated to function in its final form and under the expected conditions. Additional testing and scenarios should be conducted at this stage to cover deployment aspects, including A/B testing, blue/green deployment testing, shadow testing, and canary testing. This enables proactive and incremental testing for changing AI/ML methods and data. Before deployment, the AI system should undergo regular stress testing of both the overall system and the AI/ML components. In practice, data issues can be unpredictable. For instance, an upstream data provider may unexpectedly change its formats, or a physical event may cause a change in user behavior. Therefore, it is useful to run models in shadow mode for a period to stress test the infrastructure and assess how susceptible the AI/ML model(s) are to data-related performance degradation. AI/ML systems with data-oriented architectures are easier to test, making it simpler to detect data quality issues, data inconsistencies, and concept deviations. The pivotal decision at the conclusion of this phase is to ascertain whether and when the system should be deployed.

Level 8 Data:

If not already in place, mechanisms should be implemented to automatically log data distribution and model performance after deployment.

Level 8 Review/Test:

A comprehensive examination of all technical and product-related specifications must be undertaken, including the requisite validations.

1.1.9 Level 9 Monitoring

When using AI/ML technologies, there is a significant need to monitor the current release and explicitly consider how the next release can be optimized. For example, critical performance degradations may be hidden, or functional improvements made often have unintended consequences and limitations. At this level, the focus is on maintenance engineering - methods and pipelines for monitoring and updating AI/ML. Monitoring data quality, concept and data drift is critical. No AI system can be reliably deployed without thorough testing. For the same reason, there must be automated evaluation and reporting. When actual data [5] is available, continuous evaluation should be possible, but in many cases actual data arrives with a delay, so it is important to monitor and record model outputs to enable efficient evaluation after the fact. To this end, the AI/ML pipeline should be instrumented to log system metadata, model metadata, and the data itself.

Monitoring data quality issues and data drift is critical to detecting deviations in model behavior, especially those that are not obvious in model or product performance. Data logging is unique in the context of AI/ML systems: data logs should capture statistical properties of input functions and model predictions and record their anomalies. To monitor data, concept, and model deviations, the logs must be sent to the appropriate domain engineers. The latter is often non-trivial because the model server is not ideal for model "observability" as it does not necessarily have the right data points to connect the complex layers required for model analysis and debugging. For this reason, AI/MLTRL requires drift testing to be performed in phases well before deployment, earlier than usual. This is another reason why data-oriented-architectures should be preferred over the software industry's design-by-service approach, which makes it easier to discover and log relevant data types and sections when monitoring AI systems. To retrain and improve models, monitoring must be able to detect training biases and inform the user when retraining is required. When improving models, adding or changing features can often have unintended consequences, such as introducing latency or even bias. To minimize these risks, AI/MLTRL includes a switchback mechanism: all component or module changes to the deployed version must be reset to level 7 (integration phase) or earlier. In addition, the AI/ML of military products should provide a defined communication path for user feedback.

Level 9 Data:

Reliable AI and ML systems require adequate mechanisms for logging and verifying data, in addition to models. Systems that learn from data have unique monitoring requirements. Infrastructure and test suites should cover changes in data and environment. Military leaders should track changes in regulatory data policies.

Level 9 Review/Test:

The review/test at this stage is important for lifecycle management. Owners and stakeholders should periodically revisit this review and recommend changes as needed, as described in the Methods section. This additional monitoring during deployment helps define regulated release cycles for updated versions and provides another check for outdated model performance or other system anomalies.

1.1.10 References

[1] Alexander Lavin and Gregory Renard. Technology readiness levels for AI & ML. ICML Workshop on Challenges Deploying ML Systems, 2020.

[2] H. Zhou and Y. He. Comparative study of okr and kpi. DEStech Transactions on Economics, Business and Management, 2018.

[3] Alexander D'Amour, K. Heller, D. Moldovan, Ben Adlam, B. Alipanahi, Alex Beutel, C. Chen, Jonathan Deaton, Jacob Eisenstein, M. Hoffman, Farhad Hormozdiari, N. Houlsby, Shaobo Hou, Ghassen Jerfel, Alan Karthikesalingam, M. Lucic, Y. Ma, Cory Y. McLean, Diana Mincu, Akinori Mitani, A. Montanari, Zachary Nado, V. Natarajan, C. Nielson, Thomas F. Osborne, R. Raman, K. Ramasamy, Rory Sayres, J. Schrouff, Martin Seneviratne, Shannon Sequeira, Harini Suresh, V. Veitch, Max Vladymyrov, Xuezhi Wang, K. Webster, S. Yadlowsky, Taedong Yun, Xiaohua Zhai, and D. Sculley. Underspecification presents challenges for credibility in modern machine learning. ArXiv, abs/2011.03395, 2020.

[4] Eric Breck, Shanqing Cai, E. Nielsen, M. Salib, and D. Sculley. The ml test score: A rubric for ml production readiness and technical debt reduction. 2017 IEEE International Conference on Big Data (Big Data), pages 1123–1132, 2017.

[5] A. Botchkarev. A new typology design of performance metrics to measure errors in machine learning regression algorithms. Interdisciplinary Journal of Information, Knowledge, and Management, 14:045–076, 2019.

2. Toolkits

2.1 Security assessment toolkits for adversarial attacks

Interest in techniques and tools for evaluating the robustness of machine learning models against cyber-attacks is continuously increasing, as is the demand from customers to compare the reliability of the various available tools. Vendors, therefore, refer to security frameworks on which to base the design and construction of the product. For example, Threat Detection and Incident Response (TDIR) [2] is a strategic approach to cybersecurity management that focuses on timely identification and response to security incidents. Another framework that vendors rely on to develop their products is ModelOps[3], which specifically addresses the management and security of machine learning models in production. The combined use of Threat Detection and Incident Response with the ModelOps framework underscores the importance of proactively addressing the protection of machine learning models.
In parallel with protecting machine learning models, it is essential to consider the process of testing and validation of the models themselves, which plays a critical role in ensuring their reliability and robustness. Through a series of accurate tests and systematic validations, it is possible to identify any vulnerabilities or weaknesses in the model, allowing developers to make necessary corrections and improvements. Tests and validation procedures must be conducted at different levels, including unit tests, integration tests, and system tests, to evaluate the performance and consistency of the model in various contexts and conditions. A crucial aspect

of the testing and validation phase is also the implementation of data quality control measures to ensure that the data used to train and test the model are accurate, complete, and representative of reality. Furthermore, the adoption of standardized frameworks and methodologies for testing and validating models, such as cross-validation [4] and holdout validation [5], can facilitate comparison and comparative evaluation between different models and approaches.

2.1.1 Commercial tools

Below is a (non-exhaustive) list of commercial tools. This purely explanatory list should be updated as new tools are introduced.

- HiddenLayer ML Sec Platform[8]: a platform for evaluating the security of machine learning applications based on three modules: ML Detection and Response, ML resource integrity and vulnerability scanning, and audit reporting and vulnerability prioritization.
- Protopia Stained Glass Transform [9]: helps protect both training and inference data and provides defence tools against adversarial attacks.
- Bosch AI Platform [10]: a tool that provides protection for machine learning models, including vulnerability analysis, real-time detection, and resistance to adversarial attacks.

2.1.2 Open-source tools

Below is a (non-exhaustive) list of open-source tools. Similarly to commercial tools, this list should be updated as new tools and frameworks are introduced.

- Adversarial Robustness Toolbox (ART) [6]: an open-source library developed by IBM that provides tools for evaluating and defending machine learning models against adversarial attacks.
- TensorFlow Privacy [7]: provides tools for evaluating robustness and differential training to protect sensitive data.
- CleverHans [12]: a Python library developed for generating adversarial attacks and evaluating the robustness of machine learning models. It offers a wide range of tools and techniques for testing and improving the security of models;
- RobustDG [11]: a Python library developed by Microsoft Research to address the challenge of robustness of machine learning models in domain generalization. It offers algorithms to improve the generalization ability of models in the presence of different data distributions.

2.1.3 Evaluation process

In addition to the evaluation aspects described above, the evaluation process for assessing the trustworthiness of an AI system must not be forgotten. In principle, the procedure is as follows. Based on a risk analysis, standardized test methods are carried out according to the evaluation dimensions (see section 5.5.4.2) to obtain information regarding the trustworthiness properties with the help of suitable metrics.

2.1.4 References

[2] Red Piranha. Threat detection, investigation and response (tdir).
[3] Waldemar Hummer, Vinod Muthusamy, Thomas Rausch, Parijat Dube, Kaoutar El Maghraoui, Anupama Murthi, and Punleuk Oum. ModelOps: Cloud-based lifecycle management for reliable and trusted ai. In 2019 IEEE International Conference on Cloud Engineering (IC2E), pages 113–120. IEEE, 2019.

[4] BATES, Stephen; HASTIE, Trevor; TIBSHIRANI, Robert. Cross-validation: what does it estimate and how well does it do it?. Journal of the American Statistical Association, 2023, 1-12.
[5] NAKKIRAN, Preetum; BŁASIOK, Jarosław. The Generic Holdout: Preventing False-Discoveries in Adaptive Data Science. arXiv preprint arXiv:1809.05596, 2018.
[6] Maria-Irina Nicolae, Mathieu Sinn, Minh Ngoc Tran, Beat Buesser, Ambrish Rawat, Martin Wistuba, Valentina Zantedeschi, Nathalie Baracaldo, Bryant Chen, Heiko Ludwig, et al. Adversarial robustness toolbox v1. 0.0. arXiv preprint arXiv:1807.01069, 2018.
[7] Tensor Flow. Tensor flow privacy
[8] Hidden Layer. Hiddenlayer mlsec platform.
[9] Protopia. Stained glass transform.
[10] Bosch. Bosch ai shield.
[11] Microsoft Robust DG
[12] Cleverhans Lab. Clevarhans.

2.2 Ansys Autonomy Solution for Trustworthiness of AI in Defence

2.2.1 Preamble

Design, develop and assess systems integrating AI-algorithms for autonomy with acceptable levels of trustworthiness is a complex task and the expertise from multiple areas is required: AI know-how, systems engineering, data science, safety, security, etc. The process can be guided by regulations and standards in place, according to the application domain (e.g. aerospace, automotive), and mainly relying upon a development cycle. In the case of systems integrating AI-algorithms, the classical V-cycle is upgraded with a data management process, running all over the V-cycle phases, and also performing an inner V-cycle specific for AI design, development and validation. This finally yields a W-like cycle [1] detailed in standards like ED-324/ARP-6983[1], and ISO PAS-8800[2] (currently in progress).

2.2.2 Method for Assessment of AI Trustworthiness

The referred W-like cycle, specific to AI, is iterative and incremental and methodological support is required to follow it. The Figure 5 shows the distinctive phases of a method proposed by Ansys. The phases of the method can be conducted according to the development cycle followed in each domain (automotive or aerospace).

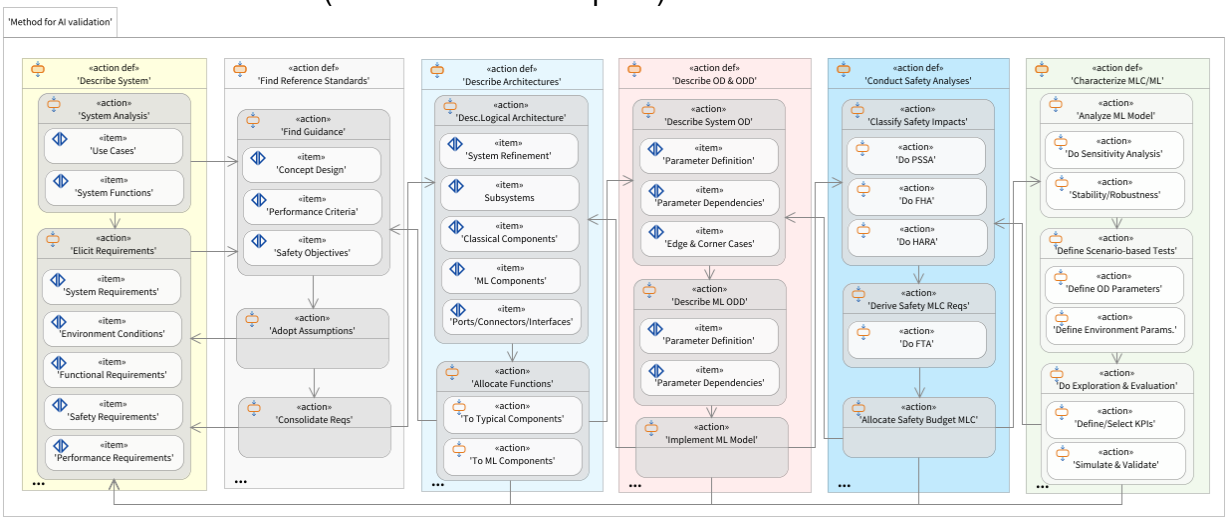

Figure 5 - Method for design, development, and validation of AI-based systems.

The proposed method provides guidance to develop the following design products:

1. ***System description.*** Analysis of Uses Cases, System Functions, and their refinements, including AI functions/components (e.g. ConOps). The analysis allows to structure requirements according to different categories: system, environment, functional, safety and performance.
2. ***Guidance for applicable standards and requirements.*** The know-how and recommended practices from existing guidelines and standards are summarized in templates which guide the design and validation of the AI system and the elicitation of requirements to be fulfilled.
3. ***Architecture Description.*** Design of the overall System Architecture including classical SW/HW and the components integrating AI-algorithms.
4. ***ODD Description.*** Description of the Operational Domain (OD) at System level, and the Operational Design Domain (ODD) at AI Component level, including ranges of ODD parameters and the data probability distributions. This phase allows analysis of the OD and ODD dependencies as well as edge and corner cases (data input singularities).
5. ***Preliminary Safety Assessment.*** As a distinctive characteristic in mission-critical systems, the Preliminary Safety Assessment helps to determine safety budgets for the overall System and derive requirements to constrain error rates produced by AI components. Techniques like PSSA, FHA, and FTA are leveraged for that purpose.
6. ***AI Algorithm Characterization.*** The characterization is conducted in two phases. The first phase covers analysis of the standalone AI algorithm, to validate sensitivity and robustness properties. The second phase relies upon scenario-based testing of the System integrating the AI algorithm, by simulation of logical and concrete scenarios involving ground truth and environment participants as per the OD and ODD. The simulation evaluates safety and autonomy KPIs and other trustworthiness indicators to ensure requirements fulfilment.
7. ***Generation of Executable Code.*** Once the AI algorithm has been characterized and fulfils its requirements, executable code can be generated from the AI model via a transformation that preserves AI model characteristics and complies with classical development standards.

2.2.3 Ansys Autonomy Solution

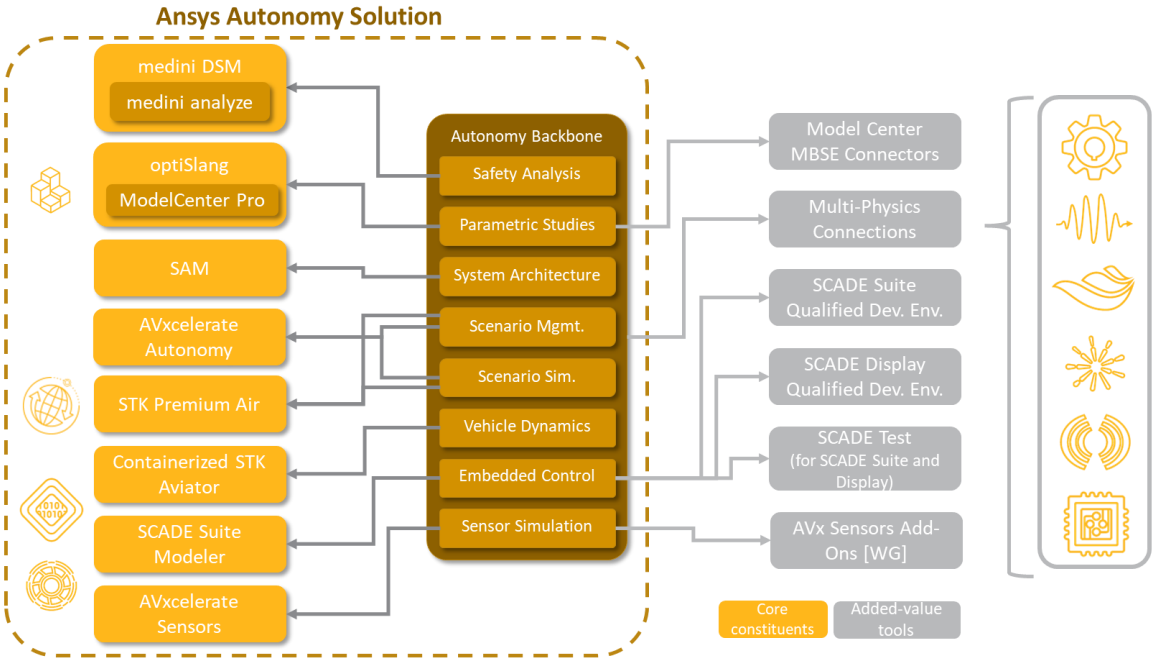

Figure 6 - Overview of the Ansys Autonomy Solution and its main constituent tools.

To support the previous method, a toolchain suite named *Ansys Autonomy Solution* is available [2]. An overview of the toolchain suite is presented in Figure 6. The Ansys Autonomy Solution is modular and thus amenable to follow methods aligned with classical V-cycles or W-like cycles, specific for AI development.

The distinctive traits of the toolchain core constituents are summarized inline:

1. ***medini analyze [3]:*** Systems, HW and SW modeler implementing analysis methods for safety (HAZOP, HARA, FHA, FTA, FMEA) and cyber-security (TOE, Attack Tree, TARA) according to standards.
2. ***SAM:*** SysML v2 [12] modeler for Systems Architecture, Use Case, Activity and Requirements engineering.
3. ***SCADE [4]:*** Environment for reliable and safe embedded SW modeling, verification, and code generation, compliant with aeronautics, automotive, railway, nuclear and general industries safety standards.
4. ***DSM [5]:*** The Digital Safety Manager drives optimization of the safety-process acting as a central hub to gather data, managing resources, planning, etc. for systems, HW and SW development projects.
5. ***STK [6]:*** The Systems Tool Kit provides a physics-based modeling and scenario-based simulation environment for analyzing platforms, physics, and payloads as they appear in real contexts of systems missions.
6. ***STK Aviator [7]:*** Aviator provides features to model and simulate aeronautical systems (aircrafts, drones) and determine their aerodynamics performance characteristics.
7. ***AVx Sensors [8]:*** Simulation engine including a catalogue for sensors-perception simulation and capabilities to test Autonomous systems.
8. ***optiSlang [9]:*** Environment for parametric designs studies, process integration & automation. Provides AI and non-AI based algorithms for e.g. Sensitivity study, Optimization or Robust Design and connects those to tool chains of engineering tools.
9. ***ModelCenter [10]:*** MBSE workflow modeler and trade-off analyzer for automation of repeatable tasks, interfaceable with most common engineering, solver, and requirement tools.

10. ***AVxcelerate Autonomy [11]:*** Cloud-based simulator, including compositional workflow and interfaces with optiSlang and AVx Sensors, able to evaluate autonomous Systems/SW performance and safety-related indicators based upon Open-Scenario format.

2.2.4 Ansys Autonomy Solution Characteristics

Generic and ad-hoc methods exist according to specific practices in different domains. Accordingly, the usage of the Ansys Autonomy Solution targets following indicators:

1. Genericity and configurability for application across different domains,
2. *Modularity* to structure tool workflows as per development cycle needs,
3. *Enabler* for assessment of *trustworthiness characteristics of AI algorithms*,
4. *Enabler* to increase *performance of W-cycles* specific for *AI development*.

2.2.5 References

[1] EASA Artificial Intelligence Concept Paper (proposed Issue 2). In https://www.easa.europa.eu/en/newsroom-and-events/news/easa-artificial-intelligence-concept-paper-proposed-issue-2-open
[2] https://www.ansys.com/
[3] *Medini analyze.* In https://www.ansys.com/products/safety-analysis/ansys-medini-analyze
[4] *SCADE. In* https://www.ansys.com/products/embedded-software/ansys-scade-suite
[5] *DSM. In* https://www.ansys.com/products/safety-analysis/ansys-digital-safety-manager
[6] *STK. In* https://www.ansys.com/products/missions/ansys-stk
[7] *STK Aviator. In* https://www.ansys.com/products/missions/ansys-stk
[8] *AVx Sensors. In* https://www.ansys.com/products/av-simulation/ansys-avxcelerate-sensors
[9] *optiSlang. In* https://www.ansys.com/products/connect/ansys-optislang
[10] *ModelCenter. In* https://www.ansys.com/products/connect/ansys-modelcenter
[11] *Avxcelerate. In* https://www.ansys.com/products/av-simulation/ansys-avxcelerate-autonomy
[12] *The Object Management Group, SysML V2 specification. In* https://www.omg.org/spec/SysML/2.0/Beta1/Language/PDF

2.3 IABG safeAI-kit to evaluate Trustworthy AI Systems

AI systems will play an increasingly significant role in future defence projects. Systematic testing and validation are therefore necessary to ensure trustworthy AI systems in military applications. Testing can also be used, for example, to identify weaknesses in the model or underlying data set, allowing developers to make improvements. IABG mbH is a product-neutral consultant in defence projects. In this context, IABG mbH develops a software toolkit to evaluate the trustworthiness of AI systems based on existing or developing relevant standards: the safeAI-kit [1].

2.3.1 How it works

The IABG safeAI-kit is developed to support the evaluation of AI systems.

Error! Reference source not found. shows the evaluation pipeline for the example of an AI system for object detection. To perform the evaluation the IABG toolkit requires one or more image datasets including annotations, black-box access to a trained model and information on the test scenarios. The black-box access is sufficient for most evaluation methods as they are based on inference results. Therefore, direct access to a model or retraining is not necessary. Nevertheless, certain methods can benefit from white-box access to provide additional evaluation details. The safeAI-kit supports the ONNX model standard, which is an open format built to represent machine learning models. ONNX allows the use and exchange of models within a variety of frameworks, tools, runtimes, and compilers.
Within the realm of AI evaluation, the IABG safeAI-kit introduces a comprehensive five-dimensional analysis which can be linked to the trustworthy properties (see section …). These dimensions encompass dataset analysis, performance and robustness evaluation, uncertainty quantification, and explainability examination, all aiming at providing a thorough understanding of AI system's behaviour and capabilities and thereby better assess the trustworthiness of AI systems.

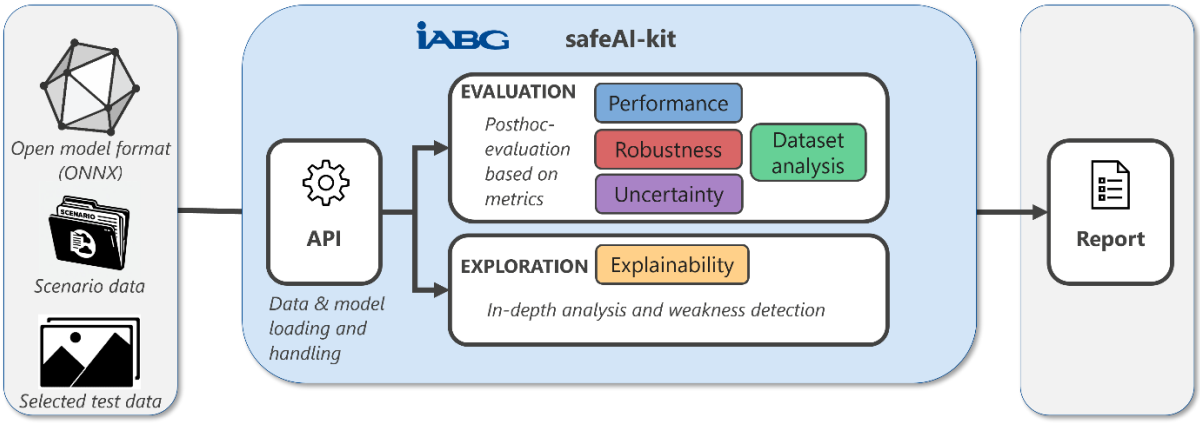

Figure 7 - IABG safe-AI kit workflow.

Finally, the evaluation results are prepared in a comprehensible manner and handed over to the customer in the form of a report. The customer can use it, for example, as a decision-making aid or, in the case of a developer, for model optimization.

2.3.2 safeAI-kit Dimensions

The subsequent section delineates the dimensions of the safeAI-kit, which are consistently undergoing refinement and expansion.

2.3.3 Dataset Analysis

This dimension delves into the balance and appropriateness of the test and training datasets. It aims to determine how well-suited the data is to the given scenario and application context. Furthermore, the data quality is assessed through tailored checks.
This toolkit enables to analyse datasets regarding the aforementioned AI development properties explained in the trustworthiness property annex:
dataset consistency, bias, dataset completeness, representativeness, balance, and data accuracy.

2.3.4 Performance Evaluation.

Performance evaluation encompasses assessing how effectively the model executes its tasks. Additionally, it enables comparative analysis, allowing to measure the model's capabilities

against other models. The performance metrics calculated in this dimension like accuracy, precision and recall among others are means to evaluate the model engineering and development properties like generalization and reliability.

2.3.5 Robustness Evaluation

Robustness testing explores the model's resilience to various data input perturbations and deviations. It assesses the model's ability to maintain performance under challenging conditions. This dimension is closely linked to the Model Engineering and Development Properties:
Stability, robustness, and repeatability

2.3.6 Explainability

The explainability analysis investigates the model's interpretability. It aims to determine whether the model's predictions and decisions can be explained and understood. Creating transparent insights to build trust, ensure fairness and address ethical concerns. It therefore considers the following beforementioned properties:
Explainability, transparency and ethical properties

2.3.7 Uncertainty Quantification

The real world is complex, chaotic, dynamically changing, and thus difficult to represent in a training set, from which models gain knowledge. Uncertainty is, therefore, inherent in the AI system's operation. For humans, it is very natural to express uncertainty when faced with a new situation or a difficult question. We use phrases like "maybe", "probably", or "I don't know". Analogously, the goal of uncertainty quantification is to enable the AI system to signal whether they are confident about the provided output or, on the contrary, that they "don't know" and are in fact guessing [2].
Correctly quantified uncertainties can contribute to trustworthiness and lead to increased safety. The analysis of uncertainties is, therefore, an important evaluation aspect and is linked to the beforementioned properties:
Confidence, repeatability

[1] https://www.iabg.de/en/business-fields/mobility-energy/safe-ai
[2]https://www.iabg.de/geschaeftsfelder/mobilitaet-energie/safe-aiabsicherung-von-ki-basierten-systemen/blog-beitrag-zur-din