Introducing RUM: A Methodological Contribution for Engineering Trustworthy AI Components in Industrial Systems

Martin Gonzalez[*1], **Loic Cantat**[*2], **Kevin Pasini**[1]

[1] IRT SystemX, France
[2] SafenAI, France
[1] {martin.gonzalez,kevin.pasini}@irt-systemx.fr - [2] loic@safenai.io

Abstract

We introduce RUM, a unified, lifecycle-aware framework for facilitating the engineering and assessing Trustworthy AI Components (AICs). Unlike model-centric evaluations, RUM treats AICs as indivisible units whose behavior must be understood across specification, development, operation, and updating phases, in conjunction with system's engineering End2End methodological activities. This announcement paper presents a series of research articles that establish the foundation of RUM: (1) a formal argument for the atomic nature of Trustworthy AICs; (2) a structured set of novel trust metrics, many of them being of non-aggregative nature, spanning the AIC's lifecycle; and (3) an operational framework introducing *AI Blueprints* to support runtime monitoring, human-in-the-loop usage, and temporal maintainability while facilitating the evolution of AICs at different stages. RUM offers a methodological contribution, aligning with AI deployment's needs in industrial contexts.

Introduction

As AI systems transition from research prototypes to **mission-critical components** in real-world applications, the question of trustworthiness becomes central, especially in industrial settings where safety, reliability, and long-term maintainability are essential. While substantial progress has been made in evaluating individual Machine Learning (ML) models, most existing approaches remain model-centric and narrowly scoped. They fail to address the realities of deployment, where AI capabilities are present in **Trustworthy AI Components (AICs)** which incorporate software units embedding pure and statistical functions, associated datasets and calibration parameters within real-world systems, which in turn interact dynamically with **complex operational environments**. These deployed AICs, which integrate ML inference with decision logic, state management, and runtime behavior, challenge classical assumptions of modularity, testability, and transparency. Strong entanglement between AIC elements (different models, datasets, hyperparameters) prevent from replacing one element by an other with similar interface/behaviour. As such, unlike isolated models, AICs often resist clean decomposition, making them opaque to traditional verification methods and difficult to monitor or maintain once integrated with respect to associated datasets and calibration parameters. Deployment failures, stakeholder misalignments, and unpredictable behavior frequently stem from this structural indivisibility and from the absence of lifecycle-aware evaluation frameworks.

To address these challenges, we introduce **RUM**: a principled, component-centric methodological contribution for engineering and assessing AI trustworthiness of industrial AICs. RUM (Robustness, Uncertainty & Monitoring) departs from conventional model-focused views and instead treats AICs as **atomic units**, characterized not just by their performance, but by their interaction with evolving contexts, human oversight, and software systems over time. The framework, open-sourced at https://catalog.confiance.ai, supports the combination of simple and complex (e.g. statistical) trust metrics and introduces AI Blueprints: reusable, lifecycle-guided design patterns that align AIC behavior with system goals & operational constraints.

This article announces a series of articles proposing foundational contributions that underpin the RUM framework and serves as a unified survey of the contents of the different articles that will constitute the series. It is organized as follows: we will first position RUM within the current and rich ecosystem of research and industry entities focusing on engineering of trustworthy AI Systems; we will then proceed to present RUM's core contributions in a simplified manner, concentrating on surveying their relevance rather than their technical detail; and finally, we will exemplify RUM's overall actionable outcome on a Visual Inspection Use-Case.

Situating RUM in an Ecosystem for Contributing to the Engineering of Trustworthy AI Components

RUM sits within the broad and rapidly evolving domain of engineering trustworthy AI systems, which spans concerns from technical robustness and transparency to lifecycle governance and human-AI interaction. This domain is highly active, reflecting both the growing maturity of AI deployment in real-world settings and the pressing need for structured, actionable approaches to manage trustworthiness beyond academic benchmarks.

The trustworthiness of AI is closely linked to: accountability (O'Neill 2014), which can be seen as either a factor of or an alternative to trust; dependability (Avizienis

[*]These authors contributed equally.

et al. 2004), encompassing attributes like safety (Cho et al. 2019), reliability, and maintainability, is also crucial. Moreover, the Assessment List for Trustworthy AI (ALTAI 2019) outlines seven core pillars: human agency and autonomy, technical robustness and security, privacy and data governance, transparency, diversity and fairness, societal and environmental welfare, and accountability. Nevertheless, while much research focuses on algorithmic features of trustworthiness, systemic analysis remains under-explored (Mattioli et al. 2023a, 2024). The latter must be evaluated across the AI lifecycle, involving various stakeholders and quantified using metrics or accuracy indicators (Braunschweig, Gelin, and Terrier 2022). At both model and system levels, it involves robustness, effectiveness, and dependability, requiring validation of intrinsic properties like accuracy, safety, and security. Thus, AI systems must remain robust in dynamic environments, avoid causing harm, and ensure human autonomy in decision-making. Trustworthiness assessment depends on context-specific attribute selection, modelled through the Operational Design Domain (ODD) and stakeholder roles (Confiance.ai 2024).

Under different degrees of modularity assumptions, attributes may be quantitative or qualitative, and their aggregation can be challenging due to incommensurability, requiring transformations to jointly commensurable scales that reflect stakeholder preferences. Multi-Criteria Decision Aiding (MCDA) (Grabisch and Labreuche 2010) provides a framework for such evaluations, using aggregation operators to derive representative values from diverse criteria. The Confiance.ai approach extends MCDA (Mattioli et al. 2023b) through structured steps: defining trustworthiness attributes, organizing them hierarchically, ensuring commensurability by assigning Key Performance Indicators (KPIs) to atomic attributes, and aggregating these indicators using advanced functions (Mattioli, Gonzalez et al. 2024). This approach accommodates dependencies and stakeholder-weighted preferences.

The problem of evaluating AI systems has historically focused on models, defined as trainable units optimized for performance on curated tasks. However, the increasing integration of AI into real-world software architectures has created a gap between model-level evaluation and system-level deployment. AI is not just about optimizing predictive accuracy but also designing reliable, maintainable, and accountable components embedded in larger systems.

Several global initiatives have emerged, proposing generic methodological frameworks and principles to address this gap, contributing to ensure more trustworthy AI. Among them, the Guidelines for Trustworthy Artificial Intelligence (Poretschkin et al. 2023) offers a general-purpose methodological approach for responsible AI development. In France, the Confiance.ai program (Confiance.ai 2024; Mattioli, Gonzalez et al. 2024) and its Canadian counterpart Confiance.ia have been pivotal in defining industrial-grade trustworthy AI practices. Similarly, the UK's Responsible AI (RAI UK) initiative (Abeywickrama and Ramchurn 2024) contributes with a national roadmap to integrate trustworthiness into AI development. In parallel, more focused works have emerged targeting specific types of AI models. For example, Fraunhofer's framework on systematic weaknesses in vision models (Gannamaneni et al. 2025) provides a structured evaluation of ML model limitations using human-interpretable dimensions. Others concentrate on application domains, such as the Safe.trAIn project's analysis of AI safety assurance in autonomous systems (Safe.trAIn 2024). Complementing these efforts, research has also addressed specific activities in the AI lifecycle. The project (CERTAIN Project 2024) articulates a "Trust by Design" method in AI engineering, while the AI4CCAM project introduces novel testing strategies (Mazouni et al. 2025). Furthermore, as AI systems increasingly interact with human users, questions about human-AI symbiosis and cognitive integration arise such as (Panai 2025), arguing for AI alignment with human cognitive models.

The Limits of Modularity in AI Components. Classical software engineering is grounded in the principle that complex systems can be decomposed into smaller, independently verifiable modules. This principle is formalized in what (Gentile 2013) has described as a *Modularity Hypothesis*, the idea that systems composed of pure functions can be *functionally* decomposed down to atomic constituents, although "modularity" is a complex notion, and there is no consensus on what its definition should be in the research community (Sun 2023, Table 1.1). In the context of AI, however, this hypothesis is intensively challenged. While there have been research efforts to retrofit modular structures onto AI systems (Sun 2023; Golechha et al. 2025; Csordás, van Steenkiste, and Schmidhuber 2021; Pfeiffer et al. 2023; Dorrell et al. 2025; Abeywickrama and Ramchurn 2024), these approaches often rely on intensive human intervention and engineering heuristics that are not representative of how AICs are deployed in production settings. Moreover, even controlled efforts to standardize data acquisition frequently result in divergent model behavior, highlighting their brittleness and contextual dependence (Recht et al. 2019).

As a consequence, attempts to audit or certify AICs using function-level unit tests, type contracts, or common behavioral assertions will fail to capture key risks: the presence of statistical functions within AICs gives rise to *complex, non-aggregative metrics*, trustworthiness properties such as maintainability under drifts, or operator interpretability of edge cases, that cannot be reduced to, or inferred from, simple & element-wise metrics alone. This complexity calls for a shift in perspective: AICs must be treated as atomic units of analysis, characterized by their externally observable behavior and their lifecycle dynamics, rather than through internal decomposition into functional parts in a quest for a final & immovable static behaviour validation. RUM emerges from this shift as we will see in the next section.

RUM for Component-Level Trustworthiness – Overview of the Upcoming Series

RUM provides a methodological framework for structuring the design, evaluation, trust management, engineering, and evolution of AICs. It does so by focusing on real-world deployment contexts and leveraging a clear system of non-aggregatable trade-offs that cannot be combined into a single

metric. It is based on three interlocking **core principles**:

1. **Atomicity of AI Components** – AICs are treated as indivisible units, embedding both deterministic logic, statistical functions, associated datasets and calibration parameters used to construct the AIC, and with inputs and outputs defined by operational requirements, not unitary isolated metrics associated to the AIC internal elements;
2. **Lifecycle-Aware Trust Trade-Offs** – Trustworthiness in AICs emerges from multidimensional trade-offs that cannot be reduced to a single global score, and must be evaluated contextually across different lifecycle stages and stakeholder perspectives. In short, their complex & non-aggregatable nature reflect that of the AICs themselves;
3. **AI Systems are Operation-Centric** – The centre of gravity for activities to build and maintain an operational AI system shifts towards the operational part of its lifecycle. The value of new data and user feedback is essential for improving performance and expanding the ODD along the AI system's life-cycle. The challenge is therefore to define the system integrating an AIC with minimal specification ready to be deployed, with the ability to extract data and feedback to initiate the loop of incremental improvement driven by the operations teams.

We highlight that these principles share a common degree of genericity: they are relative to *classes of use-cases*. A **class of use-cases** refers in this work to a family of related AI functions that share common goals (operational purpose), problem types, data modality, human-AI interaction and solution patterns, but may differ in data contents and contextual instantiations, or domain implementations. In the context of AI systems, a Use-Case Class (UCC) defines a level of abstraction that is more general than a single deployment context, yet specific enough to support reusable design patterns and architectural templates unified by similar functional intent or technological approach, analogue to those in software development (Gamma et al. 1994) but AI-focused.

Together, these principles aim to provide a framework, as illustrated in Figure 1, for aligning the technical affordances of AI with the operational demands of complex systems. Unlike system's engineering management methodologies, which guide the end-to-end design and validation of systems, RUM focuses specifically on post-deployment observability, updates, and evolutive adaptation, offering a lightweight, component-centric layer whose objective is to complement rather than replace the *End2End Methodology* (Quintero et al. 2025).

We now present the main topics of the upcoming series of articles that form the conceptual foundation of RUM, each aligned with a core principle.

RUM's Atomic approach for conceptualizing Trustworthy AI Components. This topic in the series establishes the conceptual foundation of RUM by introducing the notion of indivisible AICs as the basic unit of trustworthiness analysis. Departing from conventional model-centric and functionally decomposed views, we support this view by formalizing an Indecomposability Principle, which posits that the presence of statistical measures within AICs inherently

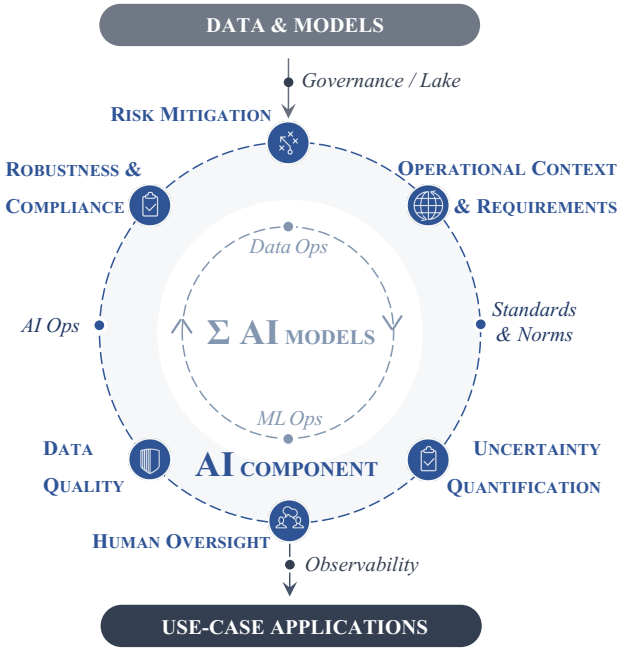

Figure 1: AI Governance and Observability to manage the deployment and operation of Trustworthy AI Components.

breaks modularity. This principle builds on Gentile's Modularity Hypothesis, asserting that when statistical functions are involved, the classical process of functional decomposition from systems engineering fails to reduce the system into isolated, pure functional blocks.

As a result, AICs, particularly those embedded in perception, prediction, or decision-making systems, exhibit non-aggregative trust characteristics that cannot be captured through analysis of submodules alone. We will provide specific examples of these in the Application section. Nevertheless, we highlight the fact that this consideration was central in the HLIF Competition feedback (Laudy et al. 2025) which ultimately concluded on the impossibility to linearly score the competition solutions without constructing a proxy of the expected fusion problem, stepping away from the reality of the fusion problem challenges. AICs must instead be treated as atomic, with their trustworthiness defined through a combination of simple metrics, as the Out-of-Distribution (OOD) ratio per class of objects, and complex & emergent metrics like performance under distribution shift, context sensitivity & reliability across scenarios. This indivisible view forms the theoretical backbone of RUM and reframes how we specify and evaluate AI behavior in real-world deployments, where we need to be able to deal with inherently complex AICs. The way AICs are structured does not follow a modular logic but rather consists of interdependent statistical elements, which co-evolve incrementally (see Fig. 2).

RUM is designed to reflect how AI is actually deployed, both in modular parts and atomic units within industry-scale AI Systems. The atomic aspect is easily visualised and understood when looking at the AIC's development cycle. As such, the take-away is that, to deliver this atomic concep-

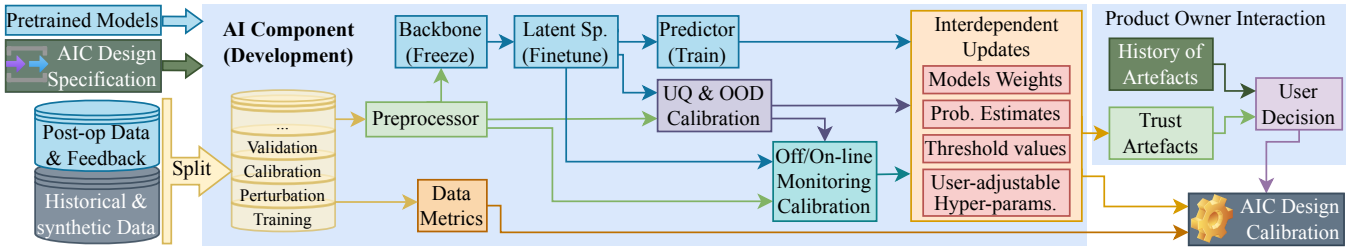

Figure 2: Development cycle of an AIC. The atomic aspect is easily exemplified: given AIC Design Specification, it consists on interdependent updates that allow to determine trade-off' values, which will be part of the AIC calibration tuning.

tualization within the broader AI system, it is necessary to integrate the AIC Design Specification heavily relying on the (Quintero et al. 2025) methodology, data, models, and supplementary algorithms to produce both the AIC and its corresponding AIC design calibration.

RUM's Contextual Trade-off and Aggregation Metrics along AI Component's Life-Cycle. Building on the atomic view of AICs established in the prior topic, this topic introduces the core evaluative framework of RUM through a structured set of trustworthiness metrics that span the entire lifecycle of an AIC. These metrics are play different roles corresponding to a different lifecycle phases: specification, development, operation, and updating. Crucially, the metric framework reflects the foundational insight that AICs embedding statistical functions cannot be exhaustively decomposed into modular, aggregable subcomponents. As such, RUM accounts for both simple metrics and a significant class of complex, non-aggregative metrics that emerge from the statistical and systemic behavior of the AIC as a whole.

Each metric set captures distinct trust trade-offs encountered in real-world engineering contexts. For example:

- Contextual uncertainty dynamics regarding a specific ODD perturbation dimension;
- Robustness estimation associated to a specific ODD zone of the overall process;
- Data temporal drift characterisation & prediction.

These will be illustrated in the Application section (Figure 7). By treating trustworthiness as a multi-stage, lifecycle-aware concern, this paper provides both a vocabulary and a quantification strategy to evaluate AICs in situ, even without full transparency into internal mechanisms. The framework supports tasks such as auditing, risk mitigation, and stakeholder communication, and is designed to extend, not replace, existing model-centric evaluation tools. It helps to identify trust issues not only at the design phase but also across deployment and evolution, which are frequently neglected in conventional assessment methods.

RUM's Operation-Centric Orientation for Actionable Trust in Runtime – Sustaining Human-in-the-Loop Confidence in Indivisible AI Components. This topic positions RUM as an operation-first methodological contribution: designed to manage the evolution of existing AICs, that are already embedded and in use within deployed AI systems, in order to provide tight feedback loops within the

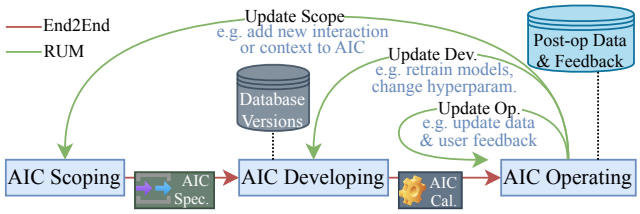

Figure 3: Cyclic evolution stages of AICs.

End2End Methodology accounting for evolving AI Systems. It introduces the concept of an *AI Blueprint*, a concept that provides ready-to-use templates for the trust engineering and evaluation of an AIC during operation, while also serving as a backward interface for iterative improvement and reuse. The genericity of such templates is neither absolute nor ad-hoc. In other words, there is no *universal* template aiming to cover all use-case typologies and there is no template that is applicable per unitary Use-Case. Instead, templates are designed per *class* of Use-Cases to tackle the curse of configuration dimensionality.

The two core ideas behind AI Blueprints are **RUM's operation-centric approach** and per **Use-Case Class (UCC) genericity** as explained earlier. On the one hand, AI systems need to evolve more frequently and easily than common software systems to be able to benefit from newly gathered data as well as operation feedback in order to improve incrementally the efficiency of the AI system. In order to do this, we need to know when and how these incremental actions shall be injected inside the End2End methodology activities (see Fig. 3). More specifically, we claim that there is a 3-stage of evolution process inside an AI system. Such evolution can be performed in operation - e.g. using only new data and feedback to allow operation teams to make evolutions - but also in development and scoping. RUM presents views for these stages and two corresponding transition artifacts as a way to define how to interact with the End2End Methodology activities around the evolution of an existing AI system. On the other hand, AI Blueprints are needed as templates for Component-level design patterns per UCC, both avoiding a curse of configuration complexity and optimizing reusability of the resulting templates.

In short, AI Blueprints' structure presents the associated Intended Purpose of the AI Function that will be specialized for a specific instance in a UCC, a conceptualization of the

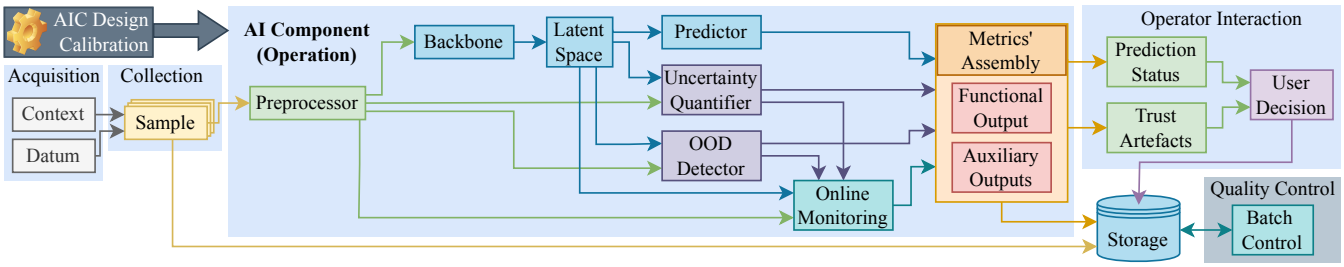

Figure 4: Operation cycle of an AIC. Provided an AIC design calibration, it interacts with different components of the AI system and the operator teams, and provides real-time & batch-wise feedback, ensures traceability and insight on AIC updates.

system's artifacts interaction with the AIC, three views reflecting the ordered stages in which RUM analyses AICs, and two new artifacts as transition elements between stages.

The first stage, illustrated in Figure 4, focuses on operation, where trust issues such as ambiguous behavior, performance drift, or misaligned operator expectations can manifest. RUM's lifecycle-aware metrics, are shown to remain fully actionable in this setting. Human operators interacting with an AIC (e.g., in decision support or visual inspection contexts) can benefit from runtime visibility into the component's uncertainty estimates, OOD flags, and system-level anomaly status. This supports a form of *in situ calibration of trust*, where feedback loops enable KPI monitoring, targeted retraining, or operational override.

The second stage focuses on development: concretely, on how RUM contributes to identifying which aspects of a deployed AIC require updates. This reverse perspective is complementary with the End2End Methodology's activities, enriching the latter by creating elements needed to compose the AIC and its associated trade-offs. RUM assumes an AIC already in operation and asks: "What needs to change in this component to sustain or improve trustworthiness?" This can include the co-update of its model weights, confidence estimation logic, monitoring mechanisms, or interpretability tools, all of which are captured in the *AI Blueprint's AIC Design Calibration* artifact. The indecomposability principle takes a further and decisive meaning in our work: it does not relate to AI/ML architectures but to trained AI/ML models: an architecture might be replaced by a different one (as replacing a ResNet backbone with a ViT) but once models are trained/finetuned/adapted within the developing stage of the AIC, they become functionally inseparable from it.

Finally, the third stage explores how RUM informs scoping decisions when transferring or adapting an AIC within a use-case class (UCC). For instance, the detection of a new source of perturbation can entail an update of the AI system's ODD or taking into consideration an observed bias in the way the operator team is exposed and interacts with the AIC's associated information. While accounting for the evolution in AIC Design Calibration is an objective in the development stage, in order to define the different degrees of freedom that the AIC possesses, one needs at least an intended purpose, an ODD, User interactions and functional performance exigencies, all captured in the *AI Blueprint's AIC Design Specification* artifact for the scoping stage. The transition decision from scope to development stage leverages the End2End Methodology to produce needed elements for stakeholders determined by the latter.

Overall, AI Blueprints serve as a comparative lens: they help to identify operational invariants and development dependencies that are likely to remain valid across all instances of a UCC. This facilitates the specification of a new AIC that is context-appropriate while maintaining alignment with trustworthiness standards. Through the structure " operation $\rightarrow$ development $\rightarrow$ scoping ", RUM highlights its distinctive positioning. It does complements the End2End Methodology by offering a practical, runtime-grounded methodological *per-UCC* contribution to maintain, evaluate, and adapt trust in AICs after they are in use. As such, RUM bridges the gap between real-time system usage and structured assurance practices, contributing toward sustained, auditable, and human-centered AI operation in industrial environments.

Applying RUM to Sustain Trust in Real-World Scenarios – A Visual Inspection Use Case

We now illustrate, the practical relevance of RUM and give a unified view of it, demonstrating the value of its key concepts on a use-case derived from a real industrial scenario. We provide non exhaustive examples of the tools at our disposal and concentrate on their use. Further quantitative and qualitative analysis will be done in the forthcoming articles.

Visual Inspection constitutes a class of use-cases characterized by the goal of automated analysis of products for quality assessment or anomaly detection using data close to image format such as ultrasound or multichannel information. That includes a wide variety of tasks such as weld inspection in automotive manufacturing, defect detection in semiconductor production, or surface analysis in food processing. These scenarios all rely on similar data structures (e.g. images), decision types (e.g. anomaly detection from different possible classes), unbalanced data and human oversight roles, allowing them to share a common blueprint structure for the AIC, despite differing in operational environment, domain constraints, or regulatory standards.

A *particular instance* within this UCC was demonstrated by an automotive industry customer, focusing specifically on weld inspection under their unique operational constraints: in one of their car manufacturing facilities, multiple visual inspection stations are positioned along each production line. Each station is equipped with cameras that capture

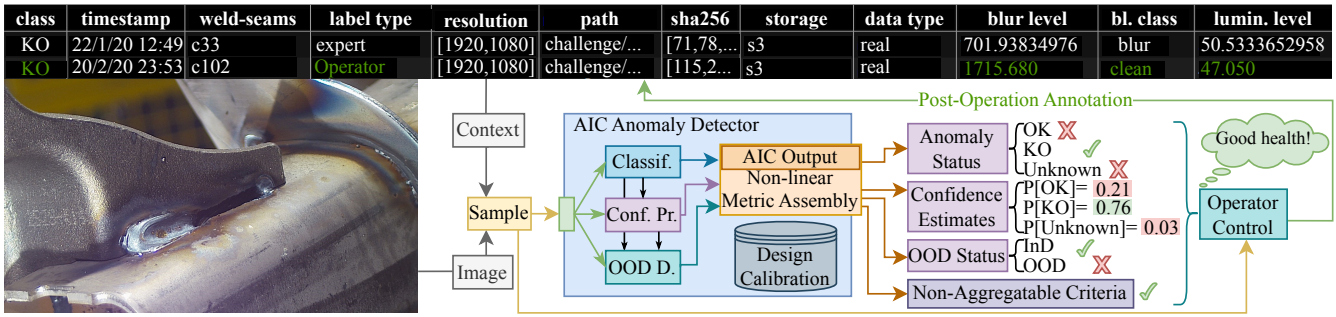

Figure 5: Sketched illustration of the exemplified AIC, its interaction with operator teams and post-operation data annotation.

images of welds on vehicle chassis. These images are sequentially reviewed by a human operator, who must decide whether each weld is acceptable or defective. On a typical production day, an operator review a total of 1,000 images, making this a high-frequency, fatigue-prone task. Error rates tend to change and usually increase over the course of a shift due to monotony and cognitive fatigue.

In our scenario, an AI system as illustrated in Figure 5 has already been integrated into production lines to assist human operators in inspecting welds on car chassis. Cameras capture images of each weld, which are then analysed by an AIC that flags potential defects. Human operators remain in the loop, making the final decision for each weld. From a ML-centric point of view, it consists on:

- A binary classifier trained using both adversarial and conformal objectives (Liu et al. 2024);
- A conformal predictor (CP) that outputs CP sets and confidence intervals. Additionally, to each label in the conformal prediction set, we associate calibrated probabilities following (Kotelevskii et al. 2025);
- An OOD detector calibrated via conformal thresholding methods (Bates et al. 2023; Novello et al. 2025);
- An Assembler that outputs an "Unknown" label when confidence is low or inputs are flagged as OOD. It combines simple metrics and non-linearly interdependent internal metrics;
- The AIC is calibrated to guarantee a 95% probability that the true class lies within the prediction set, based on stakeholder consensus.

The Visual Inspection AI Blueprint. Figure 6 illustrates the different structural components that constitute the Visual Inspection AI Blueprint. Given organic information about use-case context, it presents the following elements:

- **Intended Purpose**: This AIC is dedicated to a partially automatize anomaly detection from images using Machine Learning in order to improve productivity and quality of customer team preserving the expertise and know-how of quality control team;
- **Norms & Standards**: The normative reference for implementing the blueprint's default parameters is QIMA's Acceptable Quality Limit (AQL), associated with standard ISO 2859;

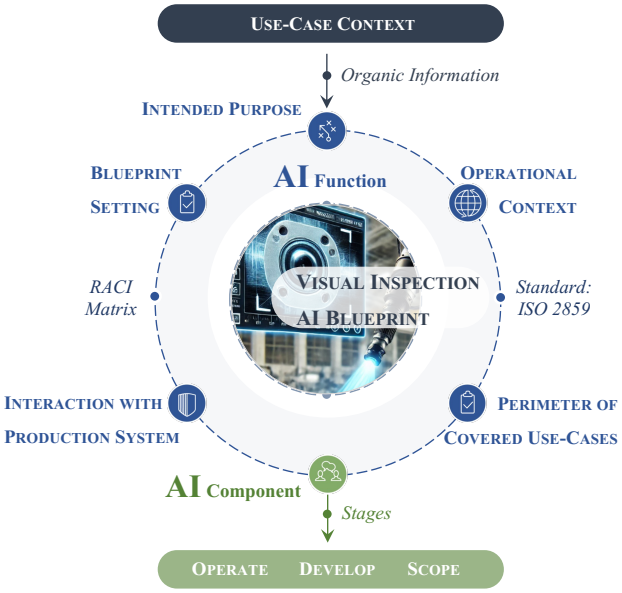

Figure 6: Visual Inspection AI Blueprint.

- **Interaction with the production system**: The blueprint interacts with the components for : data acquisition, operator interaction, quality control evaluation the AIC implementing Visual Inspection through anomaly detection.
- A setting, operational context and perimeter of covered use-cases which will be explained in detail in the corresponding paper of the series.

The Visual Inspection AI Blueprint is generalizable to a broad class of visual inspection use-cases across manufacturing, healthcare, and critical infrastructure, exemplifying the extensibility and applied relevance of RUM: maximize return on investment by aligning AI behavior with production metrics and operational requirements; maintain quality assurance through accountability and traceability of decisions regarding non-aggregative trust metrics tied to the AIC's lifecycle; support normative and regulatory compliance by ensuring traceability and stakeholder-informed control; help trustworthy human-AI collaboration by preserving accountability and oversight in operational contexts.

In sum, this use-case demonstrates how RUM supports

trustworthy AI integration at all stages of the lifecycle, from real-time use and metric-driven adjustments, to targeted development updates, to informed scoping for system redesign. The methodological contribution supports sustainable deployment by maintaining alignment between component behavior, stakeholder expectations, and system goals. Now let's take a look on the 3 different stages of the AIC supported by this Blueprint and sketch how to use them.

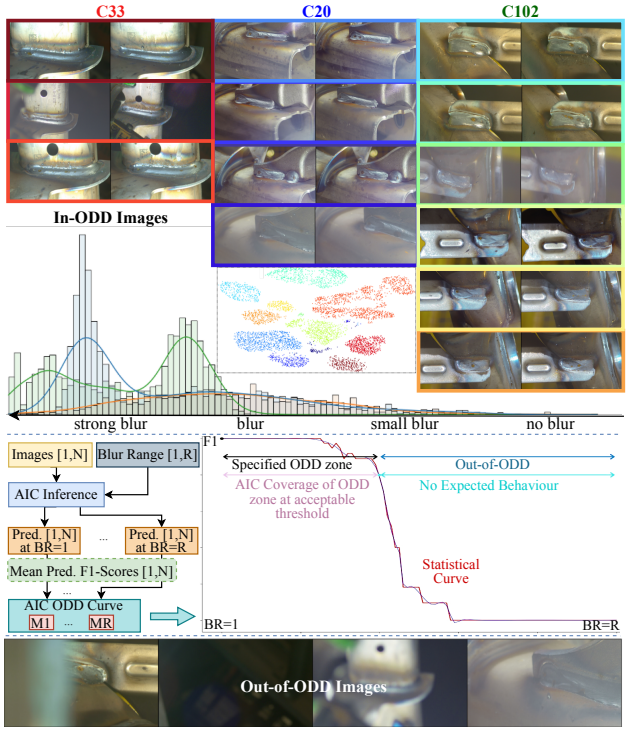

Figure 7: (Up) Latent reduction clustering & Blur distribution for valid images. (Middle) AIC Statistical Coverage of one ODD Dimension (Blur). (Down) Invalid images.

Operate Stage: Monitor and adjust the existing AIC. RUM begins with the system in operation: the deployed AIC is running in production, supporting the operator in real-time. This stage provides the operator teams with:

- The AIC outputs: a binary prediction for weld validity, a conformal prediction set and confidence estimates for each label in that set, calibrated to ensure 99% coverage, an OOD status for input images;
- A subset of images and associated real-time properties that are presented to the operator, for samples whose labelling was not automatically performed by the system,
- Per period (e.g. days) properties: drift, and aggregates;
- Information of what performance improvement can be expected from other versions of the AIC (existing or using new data captured during the prior days). RUM generates artifacts to make traceable comparisons between the active version of the component and other versions of it. This helps the operator teams to determine if the number of images presented to the operator would have been lower, or the level of uncertainty would have been better, or the number of errors reduced, etc., in order to decide to select one version of the AIC over another for the following days.

These outputs allow the operator to understand and trust the AIC's decisions, decide when to escalate cases to an expert, and flag unusual patterns. RUM enables monitoring of system health across batches of inputs and surfaces performance drifts that may arise from environmental changes (e.g., occlusions, lighting variations).

Concretely, as in Figure 7, "homogeneous" clusters are identified using HDBSCAN applied on a UMAP-reduced latent space produced by a VAE, graphically representing a data temporal drift by projecting it onto a 2D or 3D space used to display the drift and propose potential predictions such as predicting an evolution on the overall uncertainty of the AIC along a specific ODD dimension for which the AIC might have an estimated/expected robustness determined as part of the AIC design calibration. The middle statistical curve shows that the AIC covers all the specified Blur amplitude for the observed batch of images. This feedback loop allows demanded corrective action (e.g. by providing tools to better capture expert feedback, physically removing occluding objects from the data acquisition physical environment) which do not require the support of the development team (i.e. without requiring development activities). Instead, the focus is on making the AIC's behavior interpretable, steerable, and auditable using RUM's runtime trust metrics.

Develop Stage: Update the AIC to restore trustworthy performance. When operational metrics reveal issues requiring the intervention of the development team, such as degraded accuracy, frequent uncertainty, or unanticipated edge cases, RUM supports such team in formulating *targeted updates* to the AIC. Here, the development team provides an additional trade-off dimension to the operations team by adding a visual representation with a configuration item in the AIC Calibration. For example, the operation team can choose the threshold for triggering the automatic response of the AIC (without operator intervention) based on specific combination of model uncertainty level and distance from the source data. Then another trigger selector & threshold is added for a new type of welding part, since there is no reason to have the same threshold for different types of weld. This kind of modification requires a development action as opposed to a simple retraining and/or making a change in the automatic response trigger threshold for the same type of weld, which can be done without development. This might include: retraining the classifier under updated objectives or constraints (e.g., adversarial training with confidence preservation); recalibrating the conformal predictor to maintain desired error bounds; Adjusting or replacing the OOD detection module and its integration; tuning hyperparameters or revising the logic that governs how the AIC combines its metrics into outputs.

These interventions reflect deeper modifications that require coordination among developers, operators, and domain experts. RUM facilitates these updates by making the component's performance interpretable and by anchoring them

in metrics that trace their operational causes.

Scope Stage: Renew the AIC to tackle new UC's contexts. Finally, AI Blueprints support the transition of an AIC to new deployment contexts while preserving trust. This might involve: adding a new interactions & integrating the AIC with a new production system; applying the AIC to a different family of car chassis with new geometric features; adjusting the human–AI interaction loop or expanding the set of possible labels. Here, RUM enables engineers to identify structural invariants and transferable components from the existing Blueprint, guiding the design of a new, fit-for-purpose AIC. The Blueprint concept serves here as an organizational tool for aligning trust, reuse, and adaptation.

Conclusions

This paper announces a series of articles devoted to introduce RUM as a unified methodological framework for evaluating, managing and designing the trustworthiness of AICs in industrial deployment contexts. Building on existing observations around the indecomposability of AICs within AI systems, we consolidate a set of practices into a coherent methodological contribution grounded in lifecycle awareness, metric granularity, and operational alignment. RUM addresses a critical gap in the demand for methods that account for the complex, non-modular, and context-dependent nature of deployed AICs. By taking AICs as atomic units of analysis, RUM proposes tractable and actionable concepts that align with how AI is engineered, validated, and operated in industry. Beyond theoretical grounding, its operation-centric & human-in-the-loop focus, and compatibility with industrial workflows make it directly relevant to the concerns of AI engineers, quality assurance leads, system architects, and standardization and regulatory experts.

Acknowledgments

This work has been supported by the French government under the "France 2030" program, as part of IRT SystemX and within the *CSIA* project.

References

Abeywickrama, D. B.; and Ramchurn, S. D. 2024. Engineering Responsible And Explainable Models In Human-Agent Collectives. *Applied Artificial Intelligence*, 38(1).

ALTAI. 2019. Assessment List for Trustworthy Artificial Intelligence (ALTAI). Technical report, High-Level Expert Group on Artificial Intelligence, European Commission.

Avizienis, A.; et al. 2004. Basic concepts and taxonomy of dependable and secure computing. *IEEE transactions on dependable and secure computing*, 1(1): 11–33.

Bates, S.; et al. 2023. Testing for outliers with conformal p-values. *The Annals of Statistics*, 51(1): 149–178.

Braunschweig, B.; Gelin, R.; and Terrier, F. 2022. The wall of safety for AI: approaches in the Confiance.ai program. In *Proceedings of the Workshop on Artificial Intelligence Safety 2022*.

CERTAIN Project. 2024. AI Engineering for Trust by Design. Technical Report hal-04900918, HAL.

Cho, J.; et al. 2019. Stram: Measuring the trustworthiness of computer-based systems. *ACM Computing Surveys (CSUR)*, 51(6).

Confiance.ai. 2024. Towards the engineering of trustworthy AI applications for critical systems - White Paper (2nd Ed.).

Csordás, R.; van Steenkiste, S.; and Schmidhuber, J. 2021. Are Neural Nets Modular? Inspecting Functional Modularity Through Differentiable Weight Masks. In *ICLR*.

Dorrell, W.; et al. 2025. Range, not Independence, Drives Modularity in Biologically Inspired Representations. In *ICLR*.

Gamma, E.; et al. 1994. *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley Professional.

Gannamaneni, S. S.; et al. 2025. Detecting Systematic Weaknesses in Vision Models along Predefined Human-Understandable Dimensions. *preprint arXiv:2502.12360*.

Gentile, P. D. 2013. Theory of Modularity, a Hypothesis. *Procedia Computer Science*, 20: 203–209. Complex Adaptive Systems.

Golechha, S.; et al. 2025. Modular Training of Neural Networks aids Interpretability. *preprint arXiv:2502.02470*.

Grabisch, M.; and Labreuche, C. 2010. A decade of application of the Choquet and Sugeno integrals in multi-criteria decision aid. *Annals of Operations Research*, 175: 247–286.

Kotelevskii, N.; et al. 2025. Adaptive Temperature Scaling with Conformal Prediction. *preprint arXiv:2505.15437*.

Laudy, C.; Alonso, V.; Reverdy, C.; and Dreo, J. 2025. First High-Level Information Fusion Competition: Feedback and Lessons Learned. In *28th Inernational Conference on Information Fusion*.

Liu, Z.; et al. 2024. The Pitfalls and Promise of Conformal Inference Under Adversarial Attacks. In *Proceedings of the ICML*.

Mattioli, J.; Gonzalez, M.; et al. 2024. Leveraging Tropical Algebra to Assess Trustworthy AI. *Proceedings of the AAAI Symposium Series*, 4(1): 81–88.

Mattioli, J.; et al. 2023a. AI engineering to deploy reliable AI in industry. In *5th IEEE Conference on Transdisciplinary AI*.

Mattioli, J.; et al. 2023b. Towards a holistic approach for AI trustworthiness assessment based upon aids for multi-criteria aggregation. In *AAAI's Workshop on Artificial Intelligence Safety*.

Mattioli, J.; et al. 2024. An overview of key trustworthiness attributes and KPIs for trusted ML-based systems engineering. *AI and Ethics*, 4(1): 15–25.

Mazouni, Q.; et al. 2025. Mutation-Guided Metamorphic Testing of Optimality in AI Planning. *STVR*.

Novello, P.; et al. 2025. Exploring the Link Between Out-of-Distribution Detection and Conformal Prediction with Illustrations of Its Benefits. *arXiv:2403.11532*.

O'Neill, O. 2014. Trust, Trustworthiness, and Accountability. In Morris, N.; and Vines, D., eds., *Capital Failure: Rebuilding Trust in Financial Services*. Oxford University Press.

Panai, E. 2025. Transforming Artificial Intelligence into a Cognitive Extension. *Cyberpsychology, Behavior, & Social Networking*.

Pfeiffer, J.; et al. 2023. Modular Deep Learning. *TMLR*.

Poretschkin, M.; et al. 2023. Guideline for Trustworthy Artificial Intelligence: AI Assessment Catalog. *preprint arXiv:2307.03681*.

Quintero, K.; et al. 2025. An end-to-end method for operationalizing trustwothiness in AI-based critical systems. In *PESARO*.

Recht, B.; et al. 2019. Do ImageNet Classifiers Generalize to ImageNet? In *International Conference on Machine Learning*.

Safe.trAIn. 2024. Landscape of AI safety concerns – A methodology to support safety assurance for AI-based autonomous systems. *preprint arXiv:2412.14020*.

Sun, H. 2023. *Modularity in Deep Learning*. Ph.D. thesis, Université Paris-Saclay.