Information Architecture in the Age of AI

What changes for requirements - and what projects need now

AI does not make requirements engineering obsolete. It makes disciplined, iterative and traceable requirements engineering more important than ever.

AI-supported systems change the role of requirements. System behavior is no longer shaped only by specified logic, but by the interaction of business purpose, data, models, evaluation and continuous observation in production. This makes a project's information architecture a prerequisite for traceability.

Author: Reiner Schindler
Organization: MORe® Framework™
Version: V1.0 · July 2026
Publication: more-framework.org

Summary: Data does not replace requirements. It becomes a first-class requirements artifact. What matters is the connection between business need, requirement, data source, data quality, model decision, test, acceptance, monitoring and production evidence.

1. Starting point: AI changes requirements work

In traditional software projects, requirements engineering could often rely on a relatively stable assumption: a business need is described, requirements are derived from it, these requirements are implemented, tested, accepted and put into production. This sequence was never trivial, but it was basically traceable. The expected system behavior was mostly determined by specified logic.

In AI-supported systems, this basis shifts. Later system behavior is no longer produced solely by code, rules and configuration. It also emerges from data, training processes, model architectures, parameters, thresholds, evaluation methods and the actual context of use. The move from coding to training therefore also changes the demands on requirements engineering [1].

The problem is not that requirements disappear. The opposite is true. Requirements become more important because they express the business intent against which data, models and results must be judged. At the same time, traditional requirements artifacts are no longer sufficient on their own. Anyone who only describes what a system should do, without keeping trace of the data, limitations and assessments that shape this behavior, loses the business context.

The new question is not: requirements or data?

The new question is: how can business need, data basis, model behavior and production evidence remain permanently and traceably connected?

2. What AI changes in practice

AI changes requirements not at a single point, but along the entire chain of evidence. Five changes are particularly relevant.

1. System behavior becomes more probabilistic. Many AI systems do not return deterministic rule-based answers, but probabilities, classifications, recommendations or generated content. Acceptance criteria must therefore work more explicitly with quality measures, error tolerances, usage boundaries and human oversight.
2. Data becomes part of the requirements situation. The quality of the solution depends heavily on availability, provenance, representativeness, timeliness, bias risks, usage rights and data quality. Data is therefore not merely technical input, but project information with business relevance.
3. Evaluation does not replace testing, but extends it. A test case no longer only checks whether a function has been implemented correctly. It must also show how model performance, robustness, explainability, edge cases and undesirable side effects have been assessed.
4. Acceptance becomes a state. In traditional systems, acceptance often marks a point in time. In AI-supported systems, the data basis, model performance and usage context can drift. The business commitment therefore has to be observed and substantiated again after go-live.
5. Traceability extends into production. The chain of evidence does not end with the release. It must connect model version, data state, evaluation, business decision, monitoring and production observation.

Business need → Requirement → Data source → Data quality → Model decision → Acceptance criterion → Test/Evaluation → Acceptance → Release → Monitoring → Production evidence

3. Data does not replace requirements

In data-driven initiatives, it is easy to get the impression that data is the actual starting point. Operationally, this is often true: a team starts with available data, explores patterns, builds prototypes, tests model variants and approaches the business solution experimentally. For requirements engineering, however, this does not mean that requirements become secondary.

Requirements remain the expression of business intent. They answer questions such as: What problem should be solved? Which decision should be supported? What effect is desired? Which risks are acceptable? Which control must remain with humans? Which regulatory, organizational or ethical boundaries apply?

Data acts back on that intent. It can enable, refine, constrain or invalidate a requirement. For example, a business requirement may be to detect suspicious transactions earlier. Whether and how this is possible depends on available data, data quality, labeling, representativeness, historical bias, permitted use and evaluation metrics. The data situation therefore helps determine how the requirement can be formulated realistically and accepted responsibly.

> Position: Data does not replace requirements. It becomes a first-class requirements artifact. What changes are the artifacts and their connections - not the need for business structure.

Current RE4AI literature supports this view. Systematic mapping studies show that requirements engineering for AI systems is still a maturing field, and that specification, explainability, data requirements, ethics, trust and the gap between ML engineers and end users remain central challenges [2][3].

4. What is needed now

When data, models and evaluation become part of the requirements situation, projects need new artifacts. They do not necessarily have to be created in a new tool. What matters is that they are managed from a business perspective, located unambiguously and connected to the requirement.

1. Data source and data provenance. It must remain traceable which data was used, where it came from, for what purpose it was collected and whether its use is permitted.
2. Data quality and representativeness. The requirement must be connected to statements about the suitability of the data: completeness, timeliness, coverage, known gaps, bias and limitations.
3. Model decision. In AI systems, the decision layer becomes more important. It is not enough to document which model is used; it must also be clear why alternatives were rejected, thresholds were set or particular metrics were accepted.
4. Evaluation criteria. Acceptance criteria must include business-relevant quality measures: minimum performance, tolerable errors, false positives/false negatives, robustness, explainability, human override and usage boundaries.
5. Test and evaluation data. Training, validation, testing and production observation must not be mixed. It must remain traceable which data was used for which purpose.
6. Monitoring and drift observation. After go-live, it must be observed whether model performance, the data basis or the usage context change. This observation is not an operational appendix, but part of the business commitment.
7. Production evidence. In regulated environments, it is not enough to deploy a model version. It must remain traceable which business requirement became effective in production with which data state, which model version, which evaluation and which acceptance decision.

5. Information architecture instead of document archaeology

Many organizations will try to answer the new evidence requirements by creating additional documents. That is understandable, but it does not solve the underlying problem. If data descriptions, model assessments, risk decisions, acceptance criteria, test evidence and monitoring reports are managed separately, fragmentation is simply recreated.

Information architecture therefore does not mean producing more documents. It means structuring business information so that each piece of information has exactly one place and each level answers exactly one question. Documents then become views of managed information, not the leading form of that information.

For AI-supported systems, this distinction is crucial. A Datasheet for Datasets can describe which data is used and what limitations it has [4]. A Model Card can document performance, intended uses and limits of a model [5]. Both formats are valuable building blocks. But they do not replace the business-level connective structure. Only their connection with business need, requirement, decision, acceptance criterion, test, acceptance and production turns them into traceable project knowledge.

> **Traceability is not created by the individual document. It is created by the reliable connection between pieces of information.**

Using MORe® as an example, this logic can be described as follows: the business structure does not begin with the tool or with the document, but with the use case. The use case brings together - or clearly links - the need, models, rules, data reference, decisions, tests, acceptances and evidence. The structure remains method-neutral. It can support agile, traditional or hybrid approaches.

This is where configuration management comes back into play. In AI-supported systems, not only the deployed software version is configuration-relevant. Data sets, model versions, prompts, parameters, thresholds, evaluation sets, monitoring rules and operating conditions also become part of the configuration that determines the system's business effect.

Traditional configuration management often answers the question of which code or system version was productive at a given point in time. For AI systems, that is not sufficient. What matters is which business commitment was fulfilled with which data, model and system configuration. Otherwise it can no longer be reconstructed later why a system produced a certain result, which assumptions applied at the time of acceptance or which limits were known at release.

> **AI expands configuration management from technical version evidence to business evidence of system effect.**

For information architecture, this means that configuration management must not run alongside the requirement. It has to be connected with the business need, the requirement, the model decision, the acceptance criterion, the test and the production evidence. Only then can traceability cover not only what was deployed, but also why this configuration was acceptable from a business and regulatory perspective.

6. Regulatory context

Regulatory developments confirm this direction. The European Union's AI Act requires high-risk AI systems, among other things, to have data governance and data management procedures for training, validation and testing data sets, technical documentation and record-keeping obligations [6]. These requirements cannot be created reliably at the end of a project. The connection between data set, model version, business decision and evidence has to emerge during the work.

ISO/IEC 42001:2023 describes requirements for an artificial intelligence management system and addresses the establishment, implementation, maintenance and continual improvement of an AI management system [7]. The NIST AI Risk Management Framework provides a risk-based framework for organizations that design, develop, deploy or use AI systems [8].

Both approaches are valuable at the management and governance level. However, they do not automatically answer where, within a project, a business decision, a data-quality finding, an evaluation criterion or production evidence is stored and connected to the requirement.

This is where the operational gap lies.

Governance defines the expectation.

Information architecture implements it in project work.

7. Practical recommendations

Seven recommendations follow for practical work.

1. Do not start with the tool. First clarify which information answers which question. Only then decide which tool can support that structure.
2. Treat data as its own requirements artifact. Data source, provenance, quality, representativeness, usage rights and known limitations belong to the requirements situation.
3. Extend acceptance criteria. Functional criteria are not enough for AI systems. Projects need business quality measures, error boundaries, usage boundaries, explainability and human control points.
4. Document decisions where they have an effect. Why a data set, model, threshold or metric was chosen belongs in the business context, not only in meeting minutes.
5. Separate training, validation, testing and monitoring. Traceability of data states is a prerequisite for reliable evaluation and later review.
6. Do not treat acceptance as the endpoint. In AI systems, acceptance is a state that must be maintained through monitoring, drift detection and reassessment.
7. Check traceability during the work. The central control question is: can we explain today which business need is effective in production with which data, which model, which evaluation and which decision?

8. The circle closes: agility, regulation, AI

AI does not make agile ways of working obsolete. It shows whether an organization has really understood agility. Agility was never just sprint, daily, board and velocity. Agility is an organization's ability to learn under uncertainty, make decisions and translate that learning quickly into effective product development.

This ability is a prerequisite for AI-supported systems. Data has to be examined, hypotheses tested, models evaluated, feedback incorporated and business decisions adjusted iteratively. At the same time, evidence obligations, regulation and accountability increase.

Agile 1.0: deliver software iteratively.

Agile in regulated environments: deliver iteratively and remain traceable.

Agile for AI: learn iteratively, develop with data, provide regulatory evidence and monitor in production.

Each stage builds on the previous one. It adds another expectation. The method itself does not necessarily become heavier; the evidence burden does.

Anyone who understands agility only as ritual will fail with AI.

Anyone who understands agility as the capability to learn and decide needs an information architecture that keeps this learning traceable over time.

Agility without information architecture does not scale in AI projects. It only accelerates the loss of knowledge.

9. Conclusion

AI shifts requirements engineering from the description of desired functions toward the management of a business-technical evidence system. Requirements remain the expression of business intent. Data, models, evaluations and monitoring become additional artifacts that constrain, refine and substantiate that intent.

The central challenge is therefore not to produce even more documents. It is to create an information architecture in which business need, requirement, data basis, model decision, acceptance criterion, test, acceptance, release and production evidence remain permanently connected.

This makes information architecture a core capability of modern projects. In traditional projects, it reduces reconstruction. In regulated projects, it creates traceability. In AI-supported projects, it determines whether a system remains explainable, reviewable and accountable later.

Documentation is not created afterwards. It is created while working on the requirement.

References

[1] Vogelsang, A.; Borg, M.: Requirements Engineering for Machine Learning: Perspectives from Data Scientists. IEEE REW, 2019. DOI: 10.1109/REW.2019.00050.

[2] Ahmad, K.; Abdelrazek, M.; Arora, C.; Bano, M.; Grundy, J.: Requirements Engineering for Artificial Intelligence Systems: A Systematic Mapping Study. Information and Software Technology 158, 2023, 107176. DOI: 10.1016/j.infsof.2023.107176.

[3] Habiba, U. E.; Haug, M.; Bogner, J.; Wagner, S.: How mature is requirements engineering for AI-based systems? Requirements Engineering 29, 2024, pp. 567-600. DOI: 10.1007/s00766-024-00432-3.

[4] Gebru, T. et al.: Datasheets for Datasets. Communications of the ACM 64(12), 2021, pp. 86-92.

[5] Mitchell, M. et al.: Model Cards for Model Reporting. FAT* 2019, pp. 220-229.

[6] Regulation (EU) 2024/1689 (AI Act), in particular Art. 10 Data and Data Governance, Art. 11 Technical Documentation and Art. 12 Record-Keeping Obligations.

[7] ISO/IEC 42001:2023: Information technology - Artificial intelligence - Management system.

[8] NIST: Artificial Intelligence Risk Management Framework (AI RMF 1.0), NIST AI 100-1, 2023.

www.more-framework.org · mail@more-framework.org
www.projektkram.de · hallo@projektkram.de