



# Manifest

## Informationsarchitektur für agile Projekte

**Wir entdecken bessere Wege, Informationen in agilen Projekten zu managen, indem wir es selbst tun und anderen dabei helfen.**

Autor: Reiner Schindler

Organisation: MORe® Framework / projekttram

Version: v1.1 · August 2026

Quelle: [more-framework.org/framework](https://more-framework.org/framework)

*Durch diese Arbeit haben wir gelernt, Wert zu schätzen:*

**Zusammenarbeit mit Stakeholdern** **über** formale Dokumentation

**Laufende Verfeinerung von Anforderungen** **über** einmalige Spezifikationen

**Just-in-time-Analyse** **über** vollständige Vorabdefinition

**Gemeinsames Verständnis im Team** **über** individuelle Verantwortung

**Nutzbarer Mehrwert** **über** perfekte Anforderungen

Das heißt: Obwohl die Werte auf der rechten Seite wichtig sind, schätzen wir die Werte auf der linken Seite höher ein.

MORe überträgt diese agile Haltung auf die Informationsarchitektur moderner Projekte.

Alle Werte dienen einem gemeinsamen Ziel: Projektwissen nachvollziehbar zu strukturieren und dauerhaft verfügbar zu halten.

## MORe® und bestehende Standards

MORe® ersetzt keine bestehenden Standards wie IREB, arc42 oder Model Cards. MORe® verbindet ihre Artefakte über führende Orte, semantische Beziehungen und nachvollziehbare Baselines. Genau dort entsteht Traceability – nicht am Ende durch Rekonstruktion, sondern während der Arbeit.

Der Requirement Cycle steuert den Takt. Der Informationsgraph hält den fachlichen Zusammenhang.

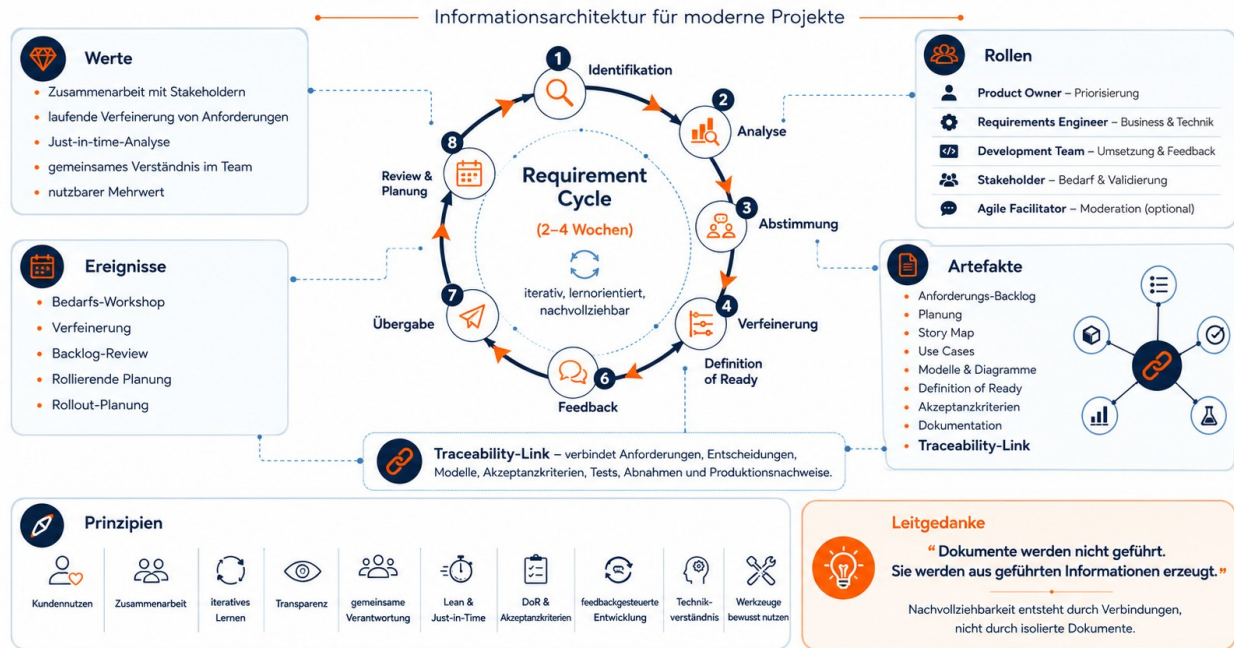


Abbildung 1: Das MORe® Framework im Überblick.

# Prinzipien

---

## 1. Kundennutzen im Fokus

Anforderungen werden entlang des geschäftlichen Mehrwerts priorisiert und regelmäßig überprüft.

## 2. Frühzeitige und kontinuierliche Zusammenarbeit

Stakeholder werden aktiv in die Anforderungsdefinition, Validierung und Verfeinerung einbezogen.

## 3. Iteratives Lernen

Anforderungen entstehen, reifen und verändern sich im Prozess. Veränderungen werden als Chance verstanden, nicht als Störung.

## 4. Transparenz

Anforderungen sind für alle Projektbeteiligten verständlich, sichtbar und nachvollziehbar dokumentiert.

## 5. Gemeinsame Verantwortung

Fachbereiche, Product Owner, Entwickler und Requirements Engineers tragen gemeinsam zum Verständnis bei.

## 6. Lean und Just-in-Time

Anforderungen werden im richtigen Detaillierungsgrad und zur richtigen Zeit ausgearbeitet. Nicht früher, nicht später.

## 7. Definition of Ready und Akzeptanzkriterien

Anforderungen sind erst dann bereit für die Umsetzung, wenn klare, überprüfbare Kriterien definiert sind.

## 8. Feedbackgesteuerte Entwicklung

Anforderungen werden regelmäßig anhand von Ergebnissen, Feedback und Daten überprüft und angepasst.

## 9. Technikverständnis fördern

Requirements Engineers verstehen technische Zusammenhänge, um Anforderungen realisierbar und testbar zu formulieren.

## 10. Werkzeuge bewusst nutzen

Tools wie Backlogs, Story Maps, Use Cases oder Modellierung unterstützen, ersetzen aber nicht das Gespräch.

# Rollen

---

MORE unterscheidet Rollen, um Klarheit über Verantwortlichkeiten im Anforderungsprozess zu schaffen.

## Product Owner

Priorisierung und Stakeholder-Sicht

Verantwortlich für die Priorisierung und geschäftliche Bewertung von Anforderungen. Bringt die Sicht der Nutzer und Stakeholder ein.

## Requirements Engineer

Bindeglied zwischen Business und Technik

Übersetzt Geschäftsbedürfnisse in verständliche, testbare und technisch realisierbare Anforderungen. Vermittelt zwischen Business und Technik.

## Development Team

Umsetzungsfeedback und Verfeinerung

Liefert Feedback zur Umsetzbarkeit und sorgt für ein gemeinsames Verständnis der Anforderungen. Unterstützt bei Verfeinerung und Umsetzung.

## Stakeholder

Anforderungen einbringen und validieren

Fachbereiche, Kunden oder Nutzer, die Anforderungen einbringen, validieren und Abnahmeentscheidungen treffen.

## Agile Facilitator

Moderation, optional

Unterstützt bei der Strukturierung und Moderation der MORE-Aktivitäten. Die Rolle kann zum Beispiel vom Scrum Master übernommen werden.

# Ereignisse

Diese Ereignisse helfen, Anforderungen kontinuierlich zu identifizieren, zu verfeinern und iterativ umzusetzen.

## Bedarfs-Workshop

Neue Anforderungen identifizieren

Ziel ist es, neue Anforderungen zu identifizieren, zu kontextualisieren und grob zu priorisieren. Stakeholder und Product Owner stehen dabei im Dialog. Die Durchführung erfolgt nach Bedarf, zum Beispiel monatlich oder zu Projektbeginn.

## Verfeinerung

Zerlegen, diskutieren, anreichern

Ziel ist es, Anforderungen zu konkretisieren, zu zerlegen, mit Akzeptanzkriterien zu versehen und technisch zu diskutieren. Die Verfeinerung erfolgt iterativ und regelmäßig, zum Beispiel wöchentlich.

## Backlog-Review

Überprüfen und umpriorisieren

Ziel ist die Überprüfung der vorhandenen Anforderungen im Backlog. Anforderungen können umpriorisiert, ergänzt, geschärft oder archiviert werden. Der Fokus liegt auf Relevanz und Reife.

## Rollierende Planung

Regelmäßige Anpassung

Ziel ist es, die Planung regelmäßig an neue Erkenntnisse, veränderte Prioritäten und den tatsächlichen Bearbeitungsstand anzupassen.

Die rollierende Planung betrachtet nicht nur den nächsten Umsetzungsschritt, sondern den fortlaufenden Bedarf an Analyse, Verfeinerung, Entscheidung, Umsetzung und Nachweis. Sie schafft Transparenz darüber, welche Anforderungen in den kommenden Requirement Cycles, Sprints, Releases oder Meilensteinen bearbeitet werden sollen.

Dabei wird die Planung nicht einmalig festgeschrieben, sondern kontinuierlich fortgeschrieben. Neue Anforderungen werden aufgenommen, bestehende Anforderungen neu priorisiert und Abhängigkeiten, Risiken oder Kapazitätsänderungen berücksichtigt.

Die rollierende Planung verbindet strategische Zielsetzung mit der operativen Arbeit an den Anforderungen.

## Rollout-Planung

Anforderungen zur Umsetzung vorbereiten

Ziel ist die abgestimmte Vorbereitung von Anforderungen, die in die Umsetzung oder Auslieferung gehen. Die Rollout-Planung bildet den Verbindungspunkt zwischen Anforderungsmanagement und Entwicklung.

# Requirement Cycle

Der Requirement Cycle ist ein wiederkehrender Zyklus von zwei bis vier Wochen zur strukturierten Arbeit mit Anforderungen.

## Die Schritte

- Identifikation
- Analyse
- Abstimmung
- Verfeinerung
- Definition of Ready
- Feedback
- Übergabe
- Abschluss mit Review und Aktualisierung der Planung

Im Abschluss wird reflektiert, welche Anforderungen abgeschlossen wurden, welche Erkenntnisse gewonnen wurden und wie sich die Prioritäten für den nächsten Cycle verändern.

## Ziele des Requirement Cycles

- Anforderungen identifizieren, analysieren und priorisieren
- Abstimmung mit Stakeholdern sicherstellen
- Anforderungen verfeinern, zum Beispiel als User Stories, Use Cases oder Modelle
- Umsetzungsreife durch Definition of Ready erreichen
- Feedback einholen, zum Beispiel durch Prototyping, Reviews oder Workshops
- Anforderungen an die Entwicklung übergeben oder zur Umsetzung freigeben

Jeder Cycle endet mit einem Review und Planning. Dabei wird reflektiert, was fertig vorbereitet ist und was im nächsten Cycle behandelt wird.

## Vorteile des Requirement Cycles

- Fördert ein iteratives, lernorientiertes Arbeiten an Anforderungen
- Verbindet strategisches Denken mit operativer Umsetzung
- Schafft Rhythmus, Struktur und Transparenz
- Erlaubt flexible Reaktion auf neue Erkenntnisse oder Änderungen

## Requirement Cycle und Scrum

Der Requirement Cycle liegt vor und neben der Entwicklung. Er identifiziert, analysiert, verfeinert und klärt Anforderungen so weit, dass daraus umsetzbare Arbeit entstehen kann.

Scrum steuert die Entwicklung. MORe® führt den fachlichen Zusammenhang. Die User Story ist dabei nicht die Spezifikation. Sie verweist auf die führende Spezifikation, die relevanten Entscheidungen, Akzeptanzkriterien, Tests und Nachweise.

# MORe® Requirement Cycle und Scrum

Wie Anforderungsarbeit und Entwicklung zusammenpassen

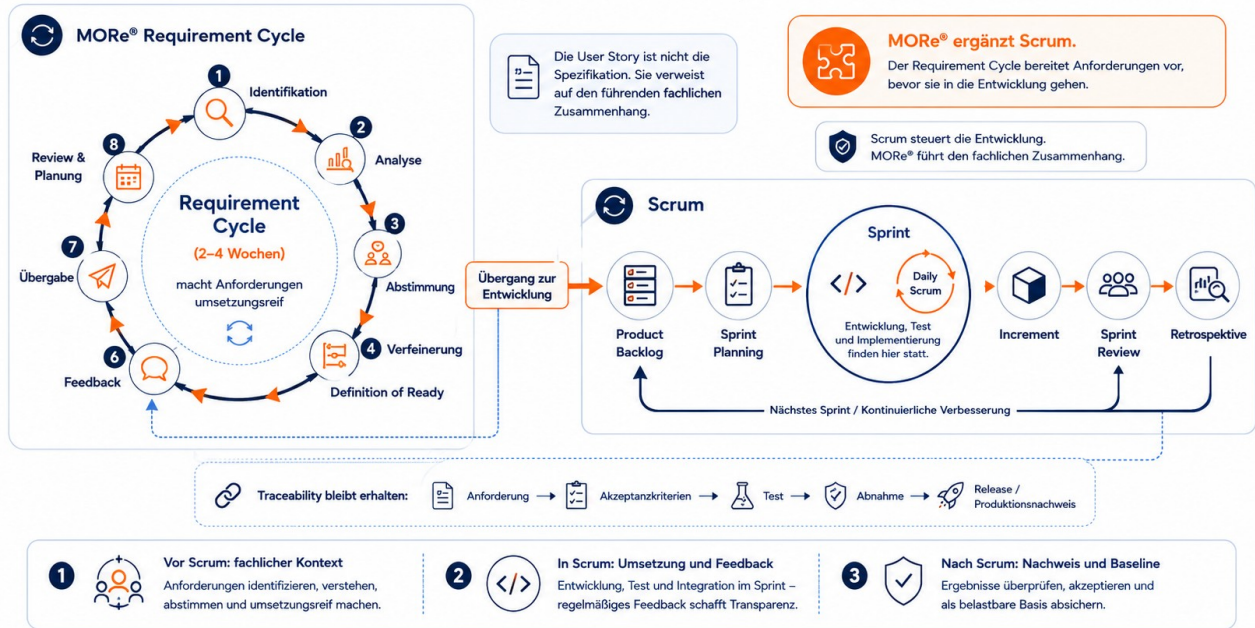


Abbildung 2: Zusammenspiel von MORe® Requirement Cycle und Scrum.



# Artefakte

MORe arbeitet mit wenigen, aber belastbaren Artefakten. Entscheidend ist nicht die Menge der Dokumentation, sondern die Nachvollziehbarkeit des Zusammenhangs.

In MORe ist nicht nur das Artefakt relevant, sondern seine Verbindung zu den anderen Artefakten.

**Dokumente werden nicht geführt. Sie werden aus geführten Informationen erzeugt.**

Ein erzeugtes Dokument ist nicht das Artefakt. Es ist lediglich eine Sicht auf das Artefakt – eine Publikationsform zu einem bestimmten Zeitpunkt, jedoch nicht der Arbeitsgegenstand selbst.

## Anforderungs-Backlog

Das Anforderungs-Backlog sammelt und priorisiert alle bekannten Anforderungen. Es schafft Transparenz darüber, welche Anforderungen existieren, welchen Status sie haben und welche als nächstes bearbeitet werden.

## Planung

Die Planung verbindet Anforderungen mit ihrer zeitlichen Umsetzung. Sie macht transparent, welche Anforderungen in welchem Requirement Cycle, Sprint, Release oder Meilenstein bearbeitet werden sollen, und wird kontinuierlich an neue Erkenntnisse angepasst. Planung ist in MORe kein einmaliges Projekt ereignis, sondern ein fortlaufender Prozess.

## Story Map

Die Story Map ordnet Anforderungen entlang von Nutzerwegen, Geschäftsprozessen oder fachlichen Abläufen. Sie hilft, Zusammenhänge zu erkennen und Anforderungen nicht isoliert zu betrachten.

## Use Cases

Use Cases beschreiben, wie Nutzer, Systeme oder Rollen mit einer Funktion interagieren. Sie machen fachliche Abläufe verständlich und helfen, Anforderungen aus konkreten Anwendungssituationen abzuleiten.

## Modelle und Diagramme

Modelle und Diagramme visualisieren Prozesse, Datenflüsse, Schnittstellen oder fachliche Strukturen. Sie reduzieren Missverständnisse und unterstützen ein gemeinsames Verständnis zwischen Fachbereich, Requirements Engineering und Entwicklung.

## Definition of Ready

MORe® verwendet eine schlanke Definition of Ready bewusst als Übergangspunkt zur Entwicklung. Sie stellt nicht sicher, dass eine User Story zur Spezifikation wird. Sie stellt sicher, dass eine User Story auf die führende Spezifikation verweist.

Damit verhindert MORe® das verbreitete Missverständnis, dass ein Ticket oder eine User Story den fachlichen Zusammenhang ersetzt. Das Ticket steuert die Umsetzung. Der fachliche Zusammenhang bleibt in der Informationsarchitektur geführt.

## Akzeptanzkriterien

Akzeptanzkriterien machen überprüfbar, wann eine Anforderung fachlich erfüllt ist. Sie verbinden Anforderungsbeschreibung, Umsetzung, Test und Abnahme.

## Dokumentation

Die Dokumentation hält fachliche Entscheidungen, Annahmen, Klärungen, Änderungen und Nachweise fest. Sie entsteht während der Arbeit an der Anforderung und nicht nachträglich als Rekonstruktion.



## Traceability-Link

Der Traceability-Link ist das verbindende Artefakt in MORe. Er verknüpft Anforderungen mit den zugehörigen Entscheidungen, Modellen, Akzeptanzkriterien, Tests, Abnahmen, Änderungen und Produktionsnachweisen.

Dadurch entsteht Nachvollziehbarkeit nicht durch einzelne Dokumente, sondern durch die belastbare Verbindung zwischen ihnen. Der Link macht sichtbar, wie aus einem fachlichen Bedarf eine umgesetzte, getestete, abgenommene und produktiv nachweisbare Änderung wurde.

## Informationsgraph und Baseline

Der Seitenbaum ist Navigation. Die fachliche Ordnung nutzt Hierarchien zur Orientierung, aber die Nachvollziehbarkeit entsteht durch semantische Beziehungen. Der Informationsgraph verbindet Anwendungsfälle und Spezifikationen mit Regeln, Entscheidungen, Akzeptanzkriterien, Tests, Baselines und – in KI-gestützten Systemen – mit Datenquellen, Modellen, Prompt/Guardrails sowie Corpus/Embedding.

Die Baseline hält den freigegebenen Stand über fachliche Artefakte, technische Umsetzung und produktiven Nachweis zusammen. Damit wird sichtbar, welche Version fachlich gilt, wodurch sie entschieden wurde, wie sie geprüft wurde und auf welcher Grundlage sie in Produktion nachweisbar bleibt.

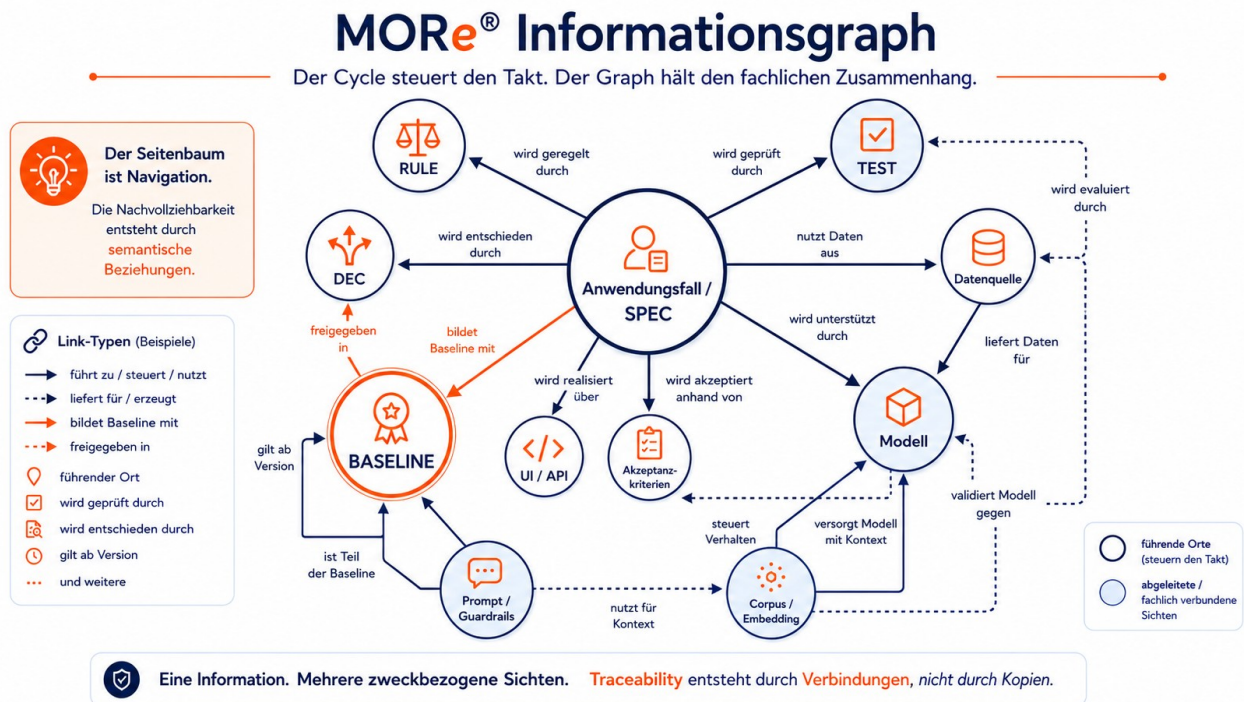


Abbildung 3: MORe® Informationsgraph mit semantischen Beziehungen, Baseline und KI-spezifischen Artefakten.

## Weiterführende Informationen

[www.more-framework.org](http://www.more-framework.org) · [mail@more-framework.org](mailto:mail@more-framework.org)

[www.projektkram.de](http://www.projektkram.de) · [hallo@projektkram.de](mailto:hallo@projektkram.de)

MORe® ist eine Marke von projektkram. Dieses Dokument gibt den Stand des Manifests zum August 2026 wieder.