

Referenzarchitektur

Wie ein Werkzeug dieser Suite gebaut ist — die Norm, aus der jede Werkzeugarchitektur abgeleitet wird

v0.4 — Entwurf zur Korrektur · Stand 2026-08-14 · dreizehn Prinzipien und zehn Verbote, alle mit Kennung und ausgewiesener Verankerung

1 · Zweck, Geltung, Abgrenzung

Diese Norm sagt, wie ein Werkzeug der Suite aufgebaut sein muss. Sie ist die Eingabe des Architektur-Agenten: Aus ihr — zusammen mit den Manifesten, dem Capability-Katalog und der Bedienungsnorm — wird die Architektur eines konkreten Werkzeugs abgeleitet.

Sie ist geerntet, nicht erfunden. Jedes Prinzip stammt entweder aus einem Manifestsatz oder aus einer Entscheidung, die in einem laufenden Produkt bereits getroffen und bewährt ist. Wo ein Prinzip keine solche Herkunft hat, ist das ausgewiesen.

Diese Norm regelt	Diese Norm regelt nicht
Den Aufbau eines Werkzeugs: Schichten, Kernobjekte, Zustände, Mandantentrennung, Nachweise.	Was ein Werkzeug fachlich tut. Das steht in der Methodennorm des Produkts.
Wo Entscheidungen fallen dürfen und wo nicht.	Wie die Oberfläche aussieht. Das steht in Erscheinungsbild und Bedienung.
Welche Bausteine verwendet werden und wie sie zusammenspielen.	Was die Bausteine im Einzelnen können. Das steht im Capability-Katalog.
Welche Nachweise ein Werkzeug erzeugen muss.	Wie geprüft wird. Das steht im Testkonzept.

2 · Die Prinzipien, ihre Kennungen und ihre Verankerung

Dreizehn Prinzipien tragen die Architektur. Jedes hat eine feste Kennung, damit spätere Dokumente auf sie verweisen können, statt sie zu wiederholen — ein Konformitätsnachweis schreibt «erfüllt RA-09» und nicht mehr «kein direkter Modellkontakt».

Die Herkunft eines Prinzips zerfällt in drei Angaben, die bisher in einer Spalte standen und Verschiedenes leisten. Sie werden in dieser Fassung getrennt geführt; was die drei bedeuten, steht in Abschnitt 2.2.

Kennung	Prinzip	Verankerung	Ableitung	Bewährt an
RA-01	Der Kern entscheidet, das Modell schlägt vor, der Mensch verantwortet	PH-03, gestützt PH-01	—	Technology-Radar (E4)
RA-02	Keine autoritativen Fakten aus dem Modell	PH-10, geschärft durch WA-02	Architekturkonzept, Faktengrenze	Invariantenprüfung PIA
RA-03	Jedes Kernobjekt trägt	PH-02	Testkonzept	—

Kennung	Prinzip	Verankerung	Ableitung	Bewährt an
	sein Governance-Mixin			
RA-04	Entwurf und Bestätigung sind getrennte Felder	PH-03	—	Divergenzprotokoll (E9), Argumente statt Werte (E14)
RA-05	Freigabe ist ein Tor; eine Änderung macht sie ungültig	PH-03	Testkonzept ADR-T04, Wissens-Lebenszyklus	—
RA-06	Änderungen sind Übergänge, keine Brüche	PH-08 (mittelbar)	Testkonzept	versionierte Bewertungen (E3, E9)
RA-07	Konfiguration vor Programmierung	keine — technische Regel	Testkonzept ADR-T05	Betriebs- und Pflegekosten je Zweig
RA-08	Mandantentrennung ist Struktur, nicht Filter	ausstehend — siehe 2.2	—	—
RA-09	Kein direkter Modellkontakt	PH-03, ausgeführt durch SC-01, SC-02	Rahmenwerk, Modellportabilität	—
RA-10	Jedes Ergebnis trägt seine Herkunft	PH-02, geschärft durch WA-02	—	—
RA-11	Unsicherheit ist ein Feld, keine Formulierung	PH-04, gestützt WA-03	—	Sicherheitsgrade (E16)
RA-12	Kontrolliertes Vokabular	keine — technische Regel	RA-05, RA-06	Kategorienwildwuchs (E7, E12)
RA-13	Ein Werkzeug erzeugt Nachweise, nicht nur Ergebnisse	PH-02	Testkonzept	—

2.1 · Die Regel für die Kennungen

Eine Kennung wird nie umgewidmet.

Kennungen sind nur dann etwas wert, wenn «erfüllt RA-09» in fünf Jahren dasselbe bedeutet wie heute. Deshalb gelten drei Regeln:

- Neue Prinzipien werden angehängt und bekommen die nächste freie Nummer. Sie werden nicht thematisch eingeschoben, auch wenn sie inhaltlich woanders hingehören würden.
- Wird ein Prinzip zurückgezogen, bleibt seine Nummer belegt und wird als zurückgezogen geführt. Sie wird nicht neu vergeben.
- Ändert sich der Inhalt eines Prinzips wesentlich, ist das ein neues Prinzip mit neuer Nummer — nicht dieselbe Nummer mit anderem Inhalt. Das alte verweist auf das neue.

Dieselben drei Regeln gelten für die Kennungen des Negativraums (NR-01 ..., Kapitel 9).

Diese Fassung nummeriert erstmals; die Reihenfolge ist deshalb noch thematisch gewählt. Ab jetzt ist sie eingefroren. Das ist dieselbe Disziplin wie bei den Invarianten des Qualitätsmodells und bei den Architekturentscheiden des Testkonzepts.

2.2 · Was Verankerung heisst — und was nicht

Drei Angaben, die Verschiedenes leisten und deshalb nicht in eine Spalte gehören:

- **Verankerung** — ein Manifestsatz. Er begründet, warum die Regel auch für ein Werkzeug gilt, das es noch nicht gibt. Nur die oberste Ebene gilt für alles; stützt sich ein Prinzip ausschliesslich auf ein Bereichsmanifest, gilt es streng genommen nur in diesem Bereich.
- **Ableitung** — eine andere Norm der Suite. Sie zeigt, woraus die Regel folgt, und trägt innerhalb der Ordnung, nicht nach aussen.
- **Bewährt an** — eine Entscheidung, die in einem Produkt getroffen und erprobt wurde. Sie ist Evidenz, nicht Begründung, und darf leer sein: Eine leere Zelle heisst «noch nirgends erprobt» und ist eine Auskunft, keine Peinlichkeit.

Ein Beleg ist keine Verankerung.

Er beantwortet nicht die Frage, warum eine Regel für ein Werkzeug gilt, das damals nicht gebaut war. Stand die Begründung eines Prinzips allein in dieser Spalte, war es keine Norm der Familie, sondern eine verallgemeinerte Einzelerfahrung — richtig vielleicht, aber vor einem Kunden nicht tragfähig.

Verankerung ist keine Deduktion.

Verlangt ist ein tragender Grund, keine formallogische Ableitung — dieselbe Regel wie bei den Manifestkennungen: Eine Kennung wird genannt, wenn sie trägt, nicht wenn sie bloss passt. Die Nähe ist deshalb verschieden; mittelbare Verankerungen sind in der Tabelle als solche gekennzeichnet.

Eine Implementierungsstrategie ist keine Ableitung.

Die Frage lautet nicht, ob ein Wert zur Regel passt, sondern ob die Regel notwendig aus ihm folgt oder bloss ein möglicher Weg dorthin ist. Konfigurierbarkeit fördert Selbstbestimmung, aber Selbstbestimmung liesse sich auch anders herstellen; deshalb wird RA-07 als technische Regel geführt und nicht verankert.

Die Kette ist nicht eins zu eins. Ein Manifestsatz kann mehrere Prinzipien tragen, ein Prinzip mehrere Werte einlösen, und eine technische Regel kann notwendig sein, ohne moralischen Gehalt zu besitzen. Ein Regelwerk, in dem ausnahmslos jede Regel aus einem Wert folgt, ist nachträglich konstruiert und nicht sauber.

Zwei Prinzipien tragen deshalb ausdrücklich keine Verankerung: RA-07 und RA-12. Eines trägt sie noch nicht: RA-08 — der Manifestsatz dazu ist formuliert, aber zurückgestellt, bis geregelt ist, wer Wissen zum Teilen bestimmen darf und was bei einem Widerruf mit dem daraus abgeleiteten Wissen geschieht.

3 · Die dreizehn Prinzipien

3.1 · RA-01 — Der Kern entscheidet, das Modell schlägt vor, der Mensch verantwortet

Drei Ebenen, die nie zusammenfallen. Das Modell erzeugt Vorschläge. Der deterministische Kern entscheidet, was gültig ist — Zustände, Zählungen, Prüfergebnisse, Freigabefähigkeit. Der Mensch entscheidet, was verbindlich wird.

Was zugesichert ist, wird deterministisch erzwungen, nicht promptseitig erbeten.

Eine Zusage, die nur im Prompt steht, wird umformuliert befolgt. Deshalb stuft der Kern eine Freigabe zurück, wenn die Voraussetzungen fehlen, und die Sprache erklärt das nur. Rückstufungen sind einseitig: Ein strengeres Urteil wird nie abgemildert.

Prüfbar: Es existiert kein Pfad, auf dem ein Modellergebnis ohne Kernprüfung verbindlich wird.

3.2 · RA-02 — Keine autoritativen Fakten aus dem Modell

Das Modell darf klassifizieren, synthetisieren, formulieren, routen und interpretieren. Es darf keine Fundstellen, Zahlen, Rechtsaussagen oder Belege erzeugen. Diese stammen aus verifizierbaren Quellen oder aus Werkzeugen.

Prüfbar: Jeder Wert mit Belegcharakter trägt eine hinterlegte Quelle; fehlt sie, ist er nicht ausgabefähig. Zählungen im Ergebnis kommen aus dem Kern, nicht aus dem Text.

Seit dem 14. August 2026 ist dieses Prinzip auf der obersten Ebene verankert: PH-10 — Beobachtetes und Vermutetes sind zweierlei; was als Tatsache auftritt, nennt seine Herkunft, und Herkunft heisst, woher etwas stammt, nicht wer es gesagt hat. Vorher stützte sich RA-02 allein auf einen bereichsgebundenen Manifestsatz und galt damit streng genommen nur dort, wo Neues von aussen hereingeholt wird.

Hinweis: Zu dieser Formulierung besteht ein offener Klärungsbedarf gegenüber dem Testkonzept (Kapitel 13, Befund 1). Diese Norm verwendet die weitere Fassung, weil die enge Fassung die fachliche Prüfung durch ein Modell ausschliessen würde.

3.3 · RA-03 — Jedes Kernobjekt trägt sein Governance-Mixin

Kernobjekte sind die Dinge, die ein Werkzeug führt und verantwortet — ein Auftrag, ein Radar-Eintrag, eine Bewertung, ein Protokoll, ein Prozessmodell. Jedes trägt dieselben Felder.

`created_at · updated_at · version · created_by · status`

Damit ist jedes Objekt datierbar, versionierbar und einem Urheber zuordenbar — Mensch oder Modell. Der Verlauf entsteht als Nebenprodukt: Wer versionierte Bewertungen speichert, bekommt die Frage «wohin hat sich die Einschätzung bewegt?» geschenkt.

Prüfbar: Kein Kernobjekt ohne vollständiges Mixin; `created_by` unterscheidet Mensch und Modell.

3.4 · RA-04 — Entwurf und Bestätigung sind getrennte Felder

Nicht nur getrennte Zustände: getrennte Felder. Der Modellentwurf bleibt neben der menschlichen Endfassung erhalten.



Abb. 1 — Aufbau eines Kernobjekts. Der Entwurf verschwindet nicht.

Zwei Gründe. Erstens ist die Differenz zwischen Entwurf und Endfassung das Divergenzprotokoll — daran lernt man das eigene Bewertungsraster und schärft die Anleitung nach. Zweitens prüft ein Mensch eine Begründung, keine Zahl; deshalb enthält der Entwurf Argumente und nicht nur Werte.

Prüfbar: Das Bestätigungsfeld für den prüfenden Menschen ist Pflicht; ohne es ist das Objekt nicht verbindlich. Der Entwurf wird nie überschrieben.

3.5 · RA-05 — Freigabe ist ein Tor, und eine Änderung macht sie ungültig



Abb. 2 — Ein Muster, das in jedem Produkt wiederkehrt.

Dieses Muster tritt in der Suite an vier Stellen auf, und es soll überall gleich funktionieren: bei der Freigabe von Wissen, bei der Freigabe eines Auftrags, beim Tor zwischen interner Validierung und Kundentest, bei der Erweiterung eines kontrollierten Vokabulars.

Freigaben kennen drei Ausprägungen, nicht zwei: freigegeben · freigegeben unter Auflage · nicht freigebbar. Die mittlere ist die praktisch wichtigste — sie erlaubt es weiterzugehen, ohne einen offenen Punkt zu verlieren, weil die Auflage mit Verantwortlichkeit und Frist dokumentiert wird.

Prüfbar: Eine Änderung an einem freigegebenen Objekt setzt den Status zurück; es gibt keinen Weg, eine Freigabe zu behalten und den Inhalt zu ändern.

3.6 · RA-06 — Änderungen sind Übergänge, keine Brüche

Die übrigen Prinzipien beschreiben, wie ein Werkzeug zu einem Zeitpunkt beschaffen ist. Dieses beschreibt, wie es sich über die Zeit verändern darf. Ohne es haben die anderen zwölf ein Verfallsdatum.

Wer eine Norm, ein Vokabular oder ein Schema ändert, ändert die Bedeutung bereits gespeicherter Daten.

Ohne festgehaltene Übergänge ist Lernen nicht möglich — weder für eine Organisation noch für einen Menschen. Wer die Bedeutung gespeicherter Werte still ändert, hat keinen Verlauf mehr, sondern nur noch einen Bestand, der aussieht wie ein Verlauf. Das ist die leiseste Art, Wissen zu zerstören: Es schlägt kein Test an, und niemand merkt es.

Bewährt am Technology-Radar: Wird dort das Bewertungsraster verschärft, ohne den Übergang festzuhalten, bedeutet dieselbe Stufe vor und nach der Änderung etwas anderes. Der Verlauf — der eigentliche Wert des Bestands — ist dann still verfälscht. Er sieht weiter gut aus und stimmt nicht mehr.

Vier Regeln halten die Zeitachse sauber:

- Jedes gespeicherte Objekt weiss, unter welcher Fassung der Norm, des Vokabulars und des Bewertungsrasters es entstanden ist. Das ist die Erweiterung des Versions-Triples von den Methoden auf die Daten.
- Jede Änderung an einer Norm liefert einen Übergang mit: Die Bestandsdaten gelten weiter, oder es gibt eine Abbildung auf die neue Fassung, oder sie werden ausdrücklich als historisch gekennzeichnet. Eine stille Neudeutung ist keine der drei Möglichkeiten.
- Das Bewertungsraster selbst ist ein versioniertes Objekt, kein Wissen im Kopf. Seine Historie gehört zum Nachweis.
- Ein ausgestellter Nachweis ist unveränderlich. Eine Korrektur ist eine neue Fassung, keine Bearbeitung der alten.

Ein Bruch bleibt erlaubt — es ist nicht alles auf Dauer verträglich zu halten. Aber er wird erklärt, datiert und auf die Bestandsdaten angewandt. Verboten ist nur der unbemerkte Bruch.

Prüfbar: Zu jedem gespeicherten Objekt lässt sich die Fassung ermitteln, unter der es entstand. Keine Änderung an einem kontrollierten Vokabular ohne Übergangsregel für Bestandsdaten. Ein ausgestellter Nachweis lässt sich nicht ändern, nur ersetzen.

Verhältnis zu RA-05: Beide schützen etwas vor unbemerkter Änderung — RA-05 den Zustand eines Objekts, RA-06 die Bedeutung seiner Werte. Und RA-06 ist die Bedingung dafür, dass RA-12 nicht zur Falle wird: Ein kontrolliertes Vokabular, das ohne Übergang erweitert wird, deutet seine eigene Vergangenheit um.

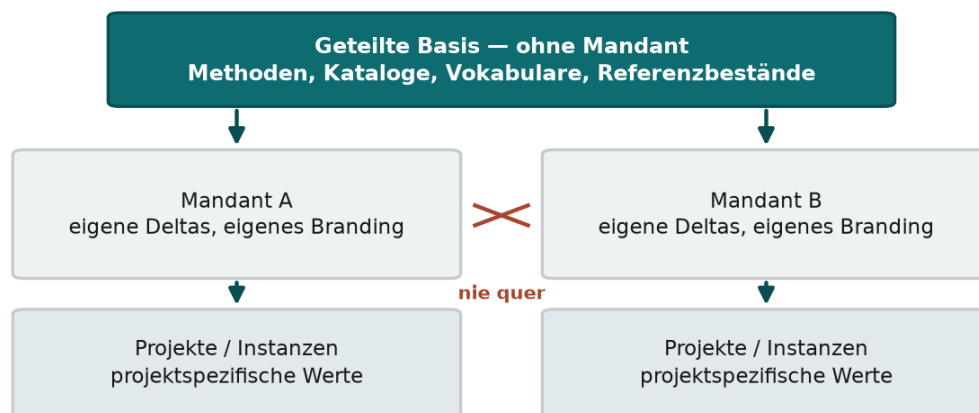
3.7 · RA-07 — Konfiguration vor Programmierung

Was sich je Mandant, je Produkt oder je Fall unterscheidet, wird nicht programmiert, sondern konfiguriert — als versionierte Datei neben dem Code, nicht in einer Datenbank und nicht in einem Fremdsystem.

Was konfiguriert wird	Warum
Methoden und Verfahren	Das fachliche WIE ändert sich schneller als der Code.
Kataloge und kontrollierte Vokabulare	Erweiterung ist ein bewusster Akt, kein Nebeneffekt.
Prüfregeln und Toleranzen	Sie sind Gegenstand fachlicher Diskussion, nicht technischer.
Testfälle	Geteiltes Werkzeug, produktspezifische Fälle.
Erscheinungsbild je Mandant	Logo und Farben sind Konfiguration, nicht Sonderbau.
Zuordnungen Funktion zu Rolle, Kostensätze	Sie unterscheiden sich je Organisation.

Prüfbar: Ein Mandantenwunsch aus dieser Liste erfordert keine Codeänderung. Alle Konfigurationsdateien sind mit dem Code versioniert und diffbar.

3.8 · RA-08 — Mandantentrennung ist Struktur, nicht Filter



Geerbt wird nach unten. Geteiltes ist ausdrücklich markiert, nicht durch Weglassen erkennbar.

Abb. 3 — Drei Ebenen; geerbt wird nach unten, nie quer.

Eine Trennung, die auf einem Filter beruht, hält nur so lange, wie niemand den Filter vergisst. Deshalb trägt jedes mandantenbezogene Objekt seine Zugehörigkeit strukturell, und Geteiltes ist ausdrücklich markiert — nicht dadurch erkennbar, dass eine Angabe fehlt.

Drei Ebenen: die geteilte Basis ohne Mandant, der Mandant mit seinen Deltas, die Instanz oder das Projekt mit seinen Werten. Geerbt wird nach unten; ein Zugriff quer über Mandantengrenzen existiert nicht als Weg, nicht als Berechtigung.

Prüfbar: Ein Zugriff ohne Mandantenbezug liefert ausschliesslich geteilte Bestände. Es gibt keine Abfrage, die zwei Mandanten zugleich sieht.

Verankerung ausstehend. Der Manifestsatz zu diesem Prinzip ist formuliert — anvertrautes Wissen bleibt getrennt, geteilt wird nur, was ausdrücklich zum Teilen bestimmt ist —, aber zurückgestellt. Solange nicht geregelt ist, wer etwas zum Teilen bestimmen darf und was bei einem Widerruf mit dem daraus abgeleiteten Wissen geschieht, stünde im Manifest ein Satz, den der laufende Betrieb bricht. Bis dahin ist RA-08 das einzige Prinzip ohne Verankerung, und es wird so ausgewiesen.

3.9 · RA-09 — Kein direkter Modellkontakt

Kein Produkt spricht direkt mit einem Modellanbieter. Alle Aufrufe — auch Einbettungen — laufen über die Capability KI-Aufruf, die je Anbieter einen Adapter führt und die Pseudonymisierung vorschaltet.

Ein Aufruf ohne Pseudonymisierung ist technisch nicht möglich, nicht bloss verboten.

Daraus folgen zwei Eigenschaften zum Preis von einer: Die Schutzschicht ist nicht umgehbar, weil es keinen zweiten Weg gibt. Und ein Modellwechsel wird zur Adapterfrage statt zur Neuentwicklung.

Prüfbar: Im Code existiert genau eine Stelle mit Anbieterkontakt. Der Aufrufnachweis hält Anbieter, Modell und Modellversion fest.

3.10 · RA-10 — Jedes Ergebnis trägt seine Herkunft

Provenienz ist Nebenprodukt der Ausführung, nicht Erinnerungsleistung des Modells. Wer die Herkunft am Schluss zusammenschreiben lässt, bekommt eine plausible Erzählung statt eines Nachweises.

Je Abschnitt eines Ergebnisses wird geführt: die verwendeten Quellenarten, die Kernaussagen mit ihrer Herkunft, die gestellten Rückfragen samt der abgelehnten, die getroffenen Annahmen und das offen Gebliebene.

Prüfbar: Die Provenienz lässt sich aus den Ausführungsdaten rekonstruieren, ohne ein Modell zu fragen. Jeder angezeigte Wert ist bis zu seinem Ursprung verfolgbar.

3.11 · RA-11 — Unsicherheit ist ein Feld, keine Formulierung

Ein Werkzeug, das seine Unsicherheit in Worte kleidet, macht sie unauswertbar. Sicherheitsgrad und Alter einer Aussage sind strukturierte Felder — bestätigt, wahrscheinlich, unbestätigt — und erscheinen in der Oberfläche, nicht nur im Datensatz.

Prüfbar: Aussagen mit Bewertungscharakter tragen einen Sicherheitsgrad; Einschätzungen tragen ihr Datum und werden als überfällig markiert, wenn sie veralten.

3.12 · RA-12 — Kontrolliertes Vokabular

Klassifiziert wird ausschliesslich in bestehende Listen. Ein Modell ordnet ein; es erweitert nicht. Neue Begriffe brauchen eine bewusste Freigabe nach dem Muster aus RA-05 und einen Übergang für die Bestandsdaten nach RA-06.

Ohne diese Regel entstehen binnen Monaten Dutzende Kategorien und die Auswertbarkeit ist verloren — die Daten sehen vollständig aus und tragen nichts mehr.

Prüfbar: Ein Wert ausserhalb der Liste wird abgewiesen, nicht stillschweigend angelegt.

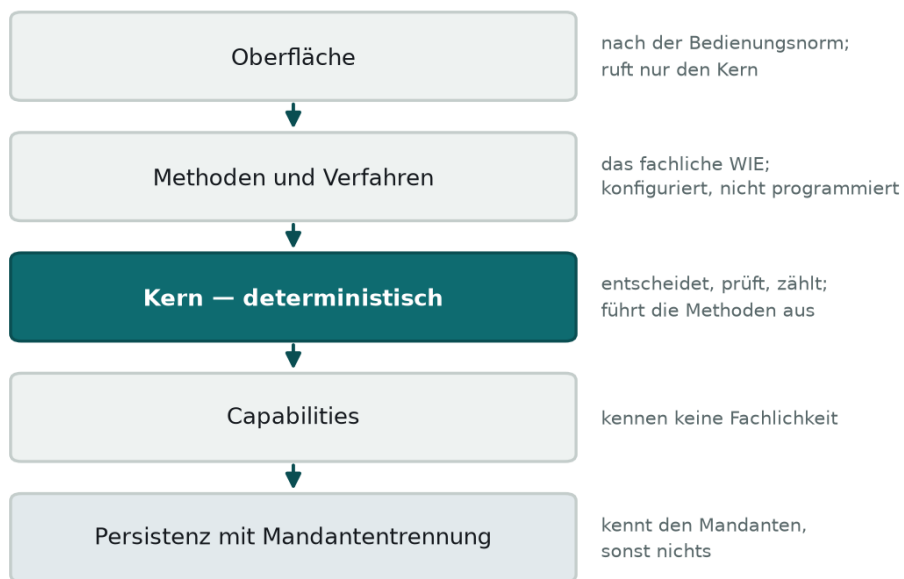
3.13 · RA-13 — Ein Werkzeug erzeugt Nachweise, nicht nur Ergebnisse

Auf die Frage «ist das geprüft?» antwortet kein Wort, sondern ein Dokument. Fünf Nachweise fallen in dieser Suite an; welche ein Werkzeug führt, hängt davon ab, was es tut.

Nachweis	Was er festhält	Wer ihn erzeugt
Provenienz	Herkunft der Aussagen eines Ergebnisses.	der Kern, bei der Ausführung
Prüfprotokoll	Ergebnis einer fachlichen Prüfung eines Ergebnisses.	die prüfende Methode
Testprotokoll	Welche Testarten mit welchem Ergebnis liefen.	der Test-Runner
Aufrufnachweis	Anbieter, Modell und Version je Modellaufruf.	die Capability KI-Aufruf
Divergenzprotokoll	Differenz zwischen Modellentwurf und menschlicher Endfassung.	der Kern, beim Bestätigen

Prüfbar: Jeder Nachweis trägt das Governance-Mixin und ist ohne das erzeugende Werkzeug lesbar.

4 · Der Schichtenaufbau



*Jede Schicht ruft nur die unmittelbar darunter.
Nur die Capability KI-Aufruf spricht mit einem Modell.*

Abb. 4 — Fünf Schichten und eine Aufrufregel.

Schicht	Was hier lebt	Was hier nicht sein darf
Oberfläche	Darstellung und Interaktion nach der Bedienungsnorm.	Fachliche Regeln, Entscheidungen, direkte Aufrufe an Capabilities.
Methoden und Verfahren	Das fachliche WIE: Verfahren, Rubriken, Kataloge, Vokabulare — konfiguriert.	Technische Bausteine, Zustandsverwaltung, Zählungen.
Kern	Domänenmodell, Zustände, Freigaben, Regelprüfung, Zählungen, Provenienz — und die Ausführung der Methoden.	Fachliches Urteil ohne Norm; anbieterspezifischer Code.
Capabilities	Technische Fähigkeiten nach Katalogvertrag.	Fachwissen jeder Art. Eine Capability weiss WIE, nicht WAS gut ist.
Persistenz	Speicherung mit struktureller Mandantentrennung.	Fachliche Logik, Zustandsübergänge.

Die Ausführungsschicht gehört in den Kern.

Was Rückfrageschleifen, Zustände, Handoffs, Abbruchkriterien und Provenienz verwaltet, ist nicht Teil einer Methode, sondern führt sie aus. Daraus folgt die Faustregel für die Aufteilung: deterministisch prüfbare Kriterien gehören in den Kern, fachlich zu beurteilende in die Methode. Diese Schicht ist im Capability-Katalog als Workflow geführt und noch nicht gebaut — sie ist die grösste offene Stelle dieser Architektur.

5 · Kernobjekte und Zustände

Ein Werkzeug führt eine überschaubare Zahl von Kernobjekten. Sie tragen das Governance-Mixin, durchlaufen den Freigabezyklus aus P5 und sind die Anker, an denen Provenienz und Nachweise hängen.

Zwei Muster haben sich bewährt und sind verbindlich:

- Das verfolgte Ding und seine Belege werden getrennt. Ein Eintrag hat eine eigene Identität; datierte Beobachtungen docken daran an. Nur so lässt sich Bewegung über die Zeit zeigen.
- Die Bewertung ist ein eigenes, versioniertes Objekt mit Zeitstempel — nicht ein Feld am bewerteten Ding. Der Verlauf entsteht dadurch kostenlos.

Prüfbar: Zu jedem Kernobjekt lässt sich beantworten, wer es wann in welchen Zustand gebracht hat und worauf es sich stützt.

6 · Oberfläche: was die Bedienungsnorm der Architektur abverlangt

Die Bedienungsnorm steht in einem eigenen Dokument. Für die Architektur ist entscheidend, dass mehrere ihrer Prinzipien nur einlösbar sind, wenn das Datenmodell sie trägt. Sie sind deshalb hier als Anforderungen aufgeführt und nicht der Oberfläche überlassen.

Prinzip der Bedienung	Was die Architektur dafür führen muss
Der Mensch entscheidet, das System schlägt vor	Entwurfs- und Bestätigungsfelder getrennt (RA-04); ein Zustand, der Entwurf von Verbindlichem unterscheidet.
Keine Aussage ohne Quelle	Herkunft je Wert abrufbar (RA-10); Quellenangabe als Pflichtfeld, nicht als

	Textkonvention.
Unsicherheit sichtbar machen	Sicherheitsgrad und Datum als Felder (RA-11); Überfälligkeit berechenbar.
Über die Familie hinweg gleich	Gleiches Status-Vokabular in allen Produkten — also im Kern, nicht je Oberfläche (RA-03, RA-12).
Vertrauen und Würde	Pseudonyme Kennungen statt Klarnamen in der Anzeige; Offenlegung als bewusster Schritt nach RA-05.

Umgekehrt gilt: Die konkreten Prüfwerte der Bedienungsnorm — Mindestgrößen, Kontraste, Farbe nie als einziges Signal — gehören in die Regelprüfung und nicht in die Sorgfalt des Entwicklers.

7 • Konfiguration und Erweiterung durch Mandanten

Die Suite verdient ihr Geld nicht daran, für jeden Kunden etwas anderes zu bauen, sondern daran, dass das Gleiche verschieden konfiguriert werden kann. Das ist eine Architekturanforderung, keine Vertriebsfrage.

Ebene	Was hier liegt	Wer es pflegt
Basis	Methoden, Kataloge, Vokabulare, Referenzbestände, Prüfregeln.	Produktverantwortung
Mandant	Deltas zur Basis: eigene Verfahren, Zuordnungen, Kostensätze, Erscheinungsbild, eigene Prüfregeln.	Beratung gemeinsam mit dem Mandanten
Instanz	Werte eines konkreten Falls.	Anwender

Regeln: Ein Mandanten-Delta überschreibt die Basis, es ersetzt sie nicht. Ein Delta ohne Zuordnung zu seinem Gegenstück in der Basis ist ungültig. Und ein Delta eines Mandanten ist für andere Mandanten nicht sichtbar — auch nicht als Vorlage.

8 • Betrieb und Umgebungen

Vier Umgebungen mit unterschiedlicher Auslegung: Entwicklung mit gemockten Schnittstellen, interne Testumgebung, Integrationsumgebung als Abbild der Produktion mit echten Schnittstellen, und der Produktivbetrieb ohne Testbetrieb.

Für die Architektur folgt daraus vor allem eines: Ein Werkzeug muss ohne Codeänderung gegen gemockte oder echte Umsysteme laufen können. Der Schnittstellenmodus ist Konfiguration und wird im Nachweis festgehalten, damit keine Scheinsicherheit entsteht.

Die Promotion läuft stufenweise und wird nie übersprungen. Jedes Produkt belegt einen eigenen Portbereich auf dem geteilten Host; Werkzeuge teilen den Betriebsstack, nicht ihre Daten.

Prüfbar: Wechsel des Schnittstellenmodus erfordert keine neue Auslieferung. Kein Produkt greift auf die Datenbestände eines anderen zu.

9 • Was ein Werkzeug nicht tun darf

Der Negativraum ist der prüfbarste Teil einer Architektur, weil Verstösse konkret sind. Jedes Verbot trägt eine Kennung, damit ein Befund sich darauf berufen kann, statt allgemein von

«Grundprinzipien» zu sprechen. Für diese Kennungen gilt dieselbe Regel wie für die Prinzipien: Sie werden nie umgewidmet.

Kennung	Ein Werkzeug darf nicht ...	Zugehöriges Prinzip
NR-01	eine Entscheidung treffen, die einem Menschen zusteht — auch nicht durch Voreinstellung, stille Automatik oder eine vorausgefüllte Wahrheit.	RA-01
NR-02	eine Fundstelle, eine Zahl oder einen Beleg erzeugen, der nicht aus einer Quelle stammt.	RA-02
NR-03	ein Modell direkt ansprechen.	RA-09
NR-04	eine Mandantengrenze überschreiten — lesend, schreibend oder als Vorlage.	RA-08
NR-05	einen freigegebenen Zustand behalten, während sich der Inhalt ändert.	RA-05
NR-06	ein kontrolliertes Vokabular eigenmächtig erweitern.	RA-12
NR-07	ein Ergebnis ohne Herkunft ausgeben.	RA-10
NR-08	eine Zusicherung allein durch Anweisung an ein Modell sicherstellen.	RA-01
NR-09	die Bedeutung gespeicherter Werte ändern, ohne den Übergang festzuhalten.	RA-06
NR-10	einen ausgestellten Nachweis nachträglich bearbeiten.	RA-06, RA-13

Ein Befund, der sich auf den Negativraum beruft, nennt die Kennung.

Ohne Kennung ist es eine Behauptung. Die Berufung auf den Negativraum wiegt in mehreren Verfahren der Suite schwer — sie macht aus einem Vorbehalt einen Muss-Befund —, und was so schwer wiegt, muss adressierbar sein.

10 · Conformance

Eine Werkzeugarchitektur ist konform, wenn sie zu jedem der dreizehn Prinzipien ausweist, wie sie es einlöst — oder begründet, warum es nicht anwendbar ist. Der Ort dieses Nachweises ist der Konformitätsnachweis aus dem Rahmenwerk.

Weil die Prinzipien Kennungen tragen, ist Liste 1 des Nachweises eine kurze Tabelle statt eines Aufsatzes — je Zeile eine Kennung, die Umsetzung und die Fundstelle. Liste 2 führt die Abweichungen mit Begründung, Liste 3 die Stellen, an denen diese Norm schweigt; die dritte Liste ist die Rückmeldung an dieses Dokument.

konformitaet:

```
- { prinzip: RA-01, erfuellt: ja, fundstelle: "Kern/Freigabe, Abschnitt 4.2" }
- { prinzip: RA-06, erfuellt: ja, fundstelle: "Schema-Version je Datensatz" }
- { prinzip: RA-08, erfuellt: nein, begruendung: "einmandantig, bewusst", liste: 2 }
- { prinzip: RA-12, erfuellt: offen, hinweis: "Vokabular noch nicht geschlossen", liste: 3
}
```

Damit wird der Nachweis maschinell prüfbar: Ein Konformitäts-Prüfer kann feststellen, ob zu jeder Kennung eine Zeile existiert — lange bevor jemand ihren Inhalt liest.

Gleiche Norm bedeutet nicht gleiches Verhalten — die Freigabe gilt je Modell und je Oberfläche.

Eine Umsetzung ist erst freigegeben, wenn sie die Pflichtinvarianten und die Ground-Truth auf dem eingesetzten Modell besteht. Der Nachweis hält Modell und Version fest.

11 · Wie ein Werkzeug aus dieser Norm entsteht

1. Zweck und Abgrenzung des Werkzeugs festhalten — was es tut und was ausdrücklich nicht.
2. Kernobjekte benennen: Was führt das Werkzeug, was ist bloss Attribut?
3. Je Kernobjekt den Freigabezyklus festlegen und die Bestätigungsfelder bestimmen.
4. Capabilities aus dem Katalog auswählen; Fehlendes als Katalogantrag ausweisen.
5. Konfigurationsgrenze ziehen: Was ist Basis, was Mandanten-Delta, was Instanzwert?
6. Nachweise festlegen: Welche der fünf erzeugt dieses Werkzeug?
7. Zeitachse festlegen: Welche Fassungen führt jedes Kernobjekt mit, und was geschieht mit Bestandsdaten, wenn sich Vokabular oder Bewertungsraster ändern?
8. Einschlägige Manifestsätze übersetzen oder als nicht anwendbar vermerken.
9. Konformitätsnachweis über alle dreizehn Kennungen erstellen.

12 · Offene Punkte dieser Norm

Punkt	Stand
Die Ausführungsschicht ist nicht gebaut.	Der Kern führt Methoden aus — bisher tut das jedes Produkt auf eigene Weise. Grösste offene Stelle.
Bestandsaufnahme der laufenden Produkte fehlt.	Diese Norm beschreibt, wie gebaut wird. Wie weit die bestehenden Produkte ihr entsprechen, ist noch nicht gemessen.
Statusvokabular ist nicht vereinheitlicht.	RA-03 verlangt gleiche Zustände über die Familie; die konkreten Werte sind produktweise gewachsen. Die Vereinheitlichung ist selbst ein Fall für RA-06.
Bestandsdaten kennen ihre Fassung noch nicht.	RA-06 verlangt, dass jedes gespeicherte Objekt seine Norm-, Vokabular- und Rasterfassung mitführt. In den laufenden Produkten ist das erst teilweise der Fall — die Nachrüstung braucht selbst einen Übergang.
Berechtigungen sind nicht geregelt.	Diese Fassung regelt die Mandamententrennung, nicht Rollen und Rechte innerhalb eines Mandanten.
Verhältnis zu bestehenden Architekturentscheiden.	Bestehende Entscheide sind hier eingearbeitet, aber noch nicht formal auf diese Norm zurückgeführt.
RA-08 trägt noch keine Verankerung.	Der Manifestsatz ist formuliert und zurückgestellt, bis Zustimmung zum Teilen und Widerruf geregelt sind. Bis dahin ausgewiesen, nicht kaschiert.
Herkunft ist bei abgeleitetem Wissen nicht transitiv.	RA-10 führt eine Angabe je Wert. Entsteht aus A und B ein C und daraus ein E, ist nach einem Widerruf von A nicht feststellbar, dass E davon abhängt. Konzeptionell verlangt das einen Provenienzgraphen. Vorgemerkt, nicht gebaut.

13 · Befunde beim Ableiten

Befund	Konsequenz
RA-08 hat keinen Manifestsatz.	Erledigt am 14.08.2026: Der Satz ist formuliert — anvertrautes Wissen bleibt getrennt — und zurückgestellt, bis Zustimmung und Widerruf geregelt sind. RA-08 wird bis dahin ohne Verankerung geführt.
RA-06 hat ebenfalls keinen Manifestsatz.	Erledigt: verankert in PH-08 (Erinnern als bleibende Fähigkeit), ausdrücklich als mittelbar gekennzeichnet.

	Die Begründung ist zugleich produktneutral gefasst worden.
Die Faktengrenze ist zweimal verschieden formuliert.	Diese Norm verwendet die weitere Fassung. PH-10 verankert das Prinzip nun auf der obersten Ebene; der Widerspruch zum Testkonzept bleibt davon unberührt, und RA-02 steht insoweit weiter unter Vorbehalt.
Drei Nachweisbegriffe waren nicht abgegrenzt.	Kapitel 3.12 grenzt sie ab. Sie bleiben getrennt, tragen aber dasselbe Governance-Mixin.
Die Bedienungsnorm stellt Anforderungen an das Datenmodell.	Kapitel 6 macht sie explizit. Ohne diese Übersetzung wären mehrere Bedienungsprinzipien nicht einlösbar, sondern nur behauptet.
Die Frage nach der Verankerung war nur zweimal gestellt worden.	Bei der Prüfung aller dreizehn Kennungen zeigte sich: fünf verankert, zwei nur bereichsgebunden, sechs ohne Antwort. Nach den Entscheiden vom 14.08.2026 sind es zehn verankert, zwei ausdrücklich technisch, eines ausstehend.
Vier von fünf Manifesten sind Produktmanifeste.	Nur die oberste Ebene gilt für alles. Solange die Bereichsmanifeste nach Werkzeugen geschnitten sind, erbt jedes darauf gestützte Prinzip deren Bereichsgrenze. Die produktneutralen Formulierungen liegen im Kennungsregister bereits vor, in den Quelldokumenten noch nicht.