

Vom Dokumentenbestand zur kontrolliert veröffentlichten Knowledge Base

Architektur eines persönlichen Wissenssystems mit offenen Formaten, Git, Obsidian, Quartz und unterstützender KI.

Ralf Seidenschwang

Fassung Juli 2026

9 ABSCHNITTE · CA. 12 MIN. LESEZEIT

ZUSAMMENFASSUNG

Persönliches und berufliches Wissen entsteht in Lebensläufen, Zeugnissen, Fachartikeln, Webseiten, Notizen und sozialen Netzwerken – meist getrennt voneinander. Das hier beschriebene System überführt ausgewählte Inhalte in eine offene, versionierbare und verknüpfte Wissensbasis.

Markdown bildet das langlebige Inhaltsformat, Obsidian die lokale Arbeitsoberfläche, Git die Änderungshistorie und Quartz die Darstellung im Web. Eine explizite Veröffentlichungs-Allowlist verhindert, dass interne Quellen oder nicht freigegebene Anhänge versehentlich in die öffentliche Fassung gelangen. KI unterstützt bei Analyse und Redaktion, entscheidet jedoch nicht autonom über Veröffentlichungen.

01 Ausgangslage und Zielsetzung

02 Wissen statt Dateisammlung

03 Realisierte Architektur

04 Datenschutz durch Freigabe

05 Automatisierung und KI

06 Praktischer Nutzen

07 Aktueller Stand und Zielbild

08 Leitlinien

09 Fazit

Ausgangslage und Zielsetzung

Dokumente lassen sich in Ordnern ablegen, doch eine Ordnerstruktur bildet Zusammenhänge nur begrenzt ab. Ein Lebenslauf, ein Arbeitszeugnis, ein Fachartikel und ein Projektbericht können dieselbe Kompetenz belegen, bleiben in einem klassischen Archiv aber getrennte Dateien. Cloubasierte Notizdienste erleichtern zwar die Suche, schaffen jedoch häufig neue Abhängigkeiten von Plattformen oder proprietären Datenformaten.

Das Projekt verfolgt daher ein anderes Ziel: Aus heterogenen Quellen soll eine nachvollziehbare Wissensarchitektur entstehen, in der Inhalte langfristig lesbar, miteinander verknüpft und für unterschiedliche Zwecke wiederverwendbar sind. Dazu zählen insbesondere:

- eine persönliche Wissens- und Dokumentationsbasis,
- die strukturierte Darstellung des beruflichen Werdegangs,
- die Aufbereitung von Fachartikeln und eigenen Beiträgen,
- ein belastbares Beispiel für technische Dokumentationskompetenz,
- eine kontrollierte Veröffentlichung ausgewählter Inhalte.

Weder ein bloßes Dateiarhiv noch eine automatisch erzeugte Bewerbungswebsite – sondern ein redaktionell gepflegter Wissensbestand mit bewusst begrenzter öffentlicher Ansicht.

Wissen statt Dateisammlung

Ein persönliches „Second Brain“ entlastet das Gedächtnis, indem es Informationen nicht nur speichert, sondern in einen nachvollziehbaren Kontext bringt. Entscheidend sind Beziehungen: Eine Berufsstation kann mit Tätigkeiten, Technologien, Veröffentlichungen und Nachweisen verbunden werden. Ein Fachbeitrag kann zugleich eine Kompetenz dokumentieren und Ausgangspunkt für weitere Beiträge sein.

Der Mehrwert entsteht durch Wiederauffindbarkeit, Verknüpfung, kontinuierliche Überarbeitung und Wiederverwendung. Wissen durchläuft deshalb einen kontrollierten Prozess:

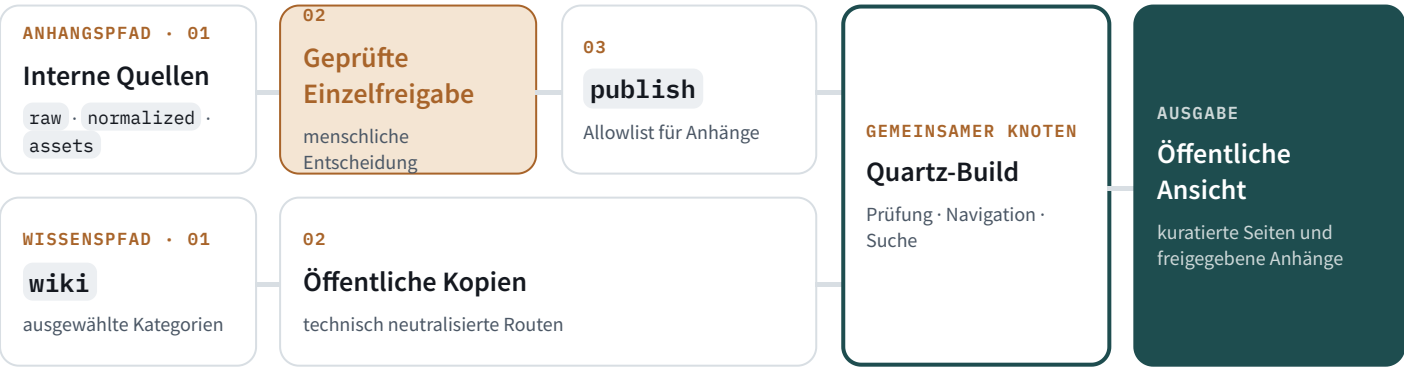


Nicht jede Quelle wird veröffentlicht. Rohmaterial und interne Arbeitsschritte bleiben erhalten, während die öffentliche Fassung nur geprüfte Inhalte enthält.

Realisierte Architektur

Die gegenwärtige Umsetzung trennt die Verarbeitung in klar definierte Schichten. Diese Trennung erleichtert Wartung und Nachvollziehbarkeit und bildet zugleich eine technische Datenschutzgrenze.

Schicht	Aufgabe	Veröffentlichung
raw	Unveränderte Originalquellen wie PDF-, DOCX-, HTML- oder EPUB-Dateien	Nicht direkt öffentlich
normalized	Aufbereitete Ableitungen, beispielsweise browserfähige Einzel-PDFs	Nur nach Einzelfreigabe
assets	Interne Bilder und Anhänge	Nur nach Einzelfreigabe
wiki	Strukturierte Wissensseiten im offenen Markdown-Format	Ausgewählte Kategorien
publish	Explizite Allowlist freigegebener Bilder und Downloads	Ausschließliche Quelle öffentlicher Anhänge
quartz	Darstellung, Navigation, Suche und statischer Web-Build	Erzeugt die öffentliche Ansicht



Wissensseiten und Anhänge nehmen getrennte Wege: Nur Anhänge durchlaufen die `publish`-Allowlist; ausgewählte Wiki-Kategorien werden als technisch neutralisierte Kopien für Quartz erzeugt.

Die vier tragenden Bausteine

Markdown & OKF

Textbasiert, maschinenlesbar und versionsfreundlich. Inhalte werden als verknüpfbare Konzepte behandelt.

Obsidian

Lokale Navigation und Bearbeitung – nicht die Datenbank. Die Inhalte bleiben gewöhnliche Dateien.

Git

Dokumentiert Änderungen. Frühere Stände sind vergleichbar und wiederherstellbar.

Quartz

Erzeugt aus der freigegebenen Auswahl eine statische, navigierbare Website.

Datenschutz durch explizite Freigabe

Ein System mit Lebensläufen und Nachweisen benötigt eine strengere Veröffentlichungslogik als ein gewöhnliches Blog. Eine Ausschlussliste wäre dafür zu fehleranfällig: Neue Dateien könnten unbemerkt öffentlich werden, wenn ihr Ausschluss vergessen wird. Die umgesetzte Lösung folgt deshalb dem umgekehrten Prinzip.

Standardmäßig bleibt alles intern; nur einzeln freigegebene Dateien werden veröffentlicht.

Die öffentliche Ansicht hat keinen direkten Zugriff auf die vollständigen Quell-, Aufbereitungs- oder Asset-Verzeichnisse. Vor jedem Build wird automatisiert geprüft, ob die freigegebenen Verweise gültig sind und innerhalb des vorgesehenen Quellbereichs bleiben. Originale werden nicht direkt veröffentlicht; personenbezogene Dokumente erscheinen ausschließlich als geprüfte, geschwärzte Fassungen. Interne Arbeitsprotokolle gehören nicht zur Website.

Technische Prüfungen reduzieren Fehler, ersetzen aber keine menschliche Freigabe. Unmittelbar vor einer externen Bereitstellung ist deshalb eine erneute manuelle Gesamtprüfung vorgesehen.

Automatisierung und KI: unterschiedliche Aufgaben

Eine glaubwürdige Architektur unterscheidet zwischen deterministischer Automatisierung und semantischer Verarbeitung. Nicht jede Aufgabe benötigt ein Sprachmodell.

Regelbasierte Automatisierung

- Dateitypen und Pfade prüfen
- Namenskonventionen validieren
- Pflichtfelder und Links kontrollieren
- Allowlist und Build ausführen
- Änderungsdatum aus Git ableiten

Geeignete KI-Unterstützung

- Dokumente zusammenfassen
- Bezüge zwischen Themen vorschlagen
- Rohtexte strukturieren und überarbeiten
- Widersprüche oder sensible Stellen markieren
- Inhalte für weitere Formate entwerfen

KI unterstützt die Erschließung und Redaktion vorhandener Informationen. Sie darf weder ungeprüfte Aussagen als Tatsachen übernehmen noch eigenständig sensible Inhalte veröffentlichen. Die KI-Schicht soll langfristig austauschbar bleiben; je nach Aufgabe kommen cloudbasierte Modelle, lokale Modelle oder vollständig regelbasierte Verarbeitung infrage.

Praktischer Nutzen

Recruiting & Fachabteilungen

Die Knowledge Base zeigt nicht nur Stationen, sondern auch, wie technische Themen erschlossen, dokumentiert und vermittelt werden.

Dokumentation & Lernen

Fehlerbilder, Lösungswege, Architekturentscheidungen und Lernergebnisse bleiben auffindbar und wiederverwendbar.

Weitere Formate

Aus dem geprüften Bestand können Fachartikel, LinkedIn-Beiträge, Präsentationen, Videoskripte oder Whitepaper entstehen.

Aktueller Stand und Zielbild

Bereits umgesetzt

- Getrennte Schichten für Originale, Aufbereitung, Wissen, Freigaben und Darstellung
- Strukturierte Wissensseiten für Werdegang, Lebensläufe, Nachweise, Veröffentlichungen und Referenzen
- Lokale Bearbeitung mit Obsidian und Versionierung mit Git
- Quartz-basierte lokale Webansicht
- Explizite Allowlist mit automatischer Prüfung vor jedem Build
- Automatisch erzeugter Aktualitätsnachweis
- Manuell befüllter, nicht veröffentlichter Eingangsbereich
- Datenschutzprüfung und Schwärzung der freigegebenen Zeugnisfassungen
- Erste Prüfung der lokalen Git-Historie auf sensible Inhalte

Geplante Weiterentwicklung

- Erneute Abschlussprüfung unmittelbar vor der externen Bereitstellung
- Veröffentlichung unter `profil.ralfsei.de`
- Netlify-Deployment und DNS für die Veröffentlichung einrichten
- Produktivtest von HTTPS, Links, Downloads und Sitemap
- Optional stärkere Import- und Qualitätsautomatisierung
- Optional zielgruppenspezifische Ansichten
- Perspektivisch austauschbare lokale oder cloudbasierte KI-Unterstützung

Hinweis: Die geplanten Punkte sind kein Bestandteil des heutigen Produktivstands. Insbesondere findet keine automatische Veröffentlichung neuer Eingangsdokumente statt.

Leitlinien für die Weiterentwicklung

01 Intern ist der Standard.

Inhalte werden nur nach bewusster Prüfung öffentlich.

02 Offene Formate sichern Nachhaltigkeit.

Wissen bleibt unabhängig von einzelnen Werkzeugen lesbar.

03 Regeln haben Vorrang vor KI.

Eindeutige Prüfungen werden deterministisch umgesetzt.

04 KI berät, der Mensch entscheidet.

Unsichere Fälle führen zu Rückfragen, nicht zu automatischer Freigabe.

05 Quellen und Darstellung bleiben getrennt.

Die Website ist eine kontrollierte Sicht, nicht der gesamte Wissensbestand.

06 Nachvollziehbarkeit ist Teil der Qualität.

Änderungen, Freigaben und Entscheidungen bleiben überprüfbar.

Fazit

Das Projekt verbindet persönliches Wissensmanagement, technische Dokumentation und professionelle Außendarstellung. Seine Besonderheit liegt nicht in einem einzelnen Werkzeug, sondern im Zusammenspiel offener Inhalte, klarer Schichten, nachvollziehbarer Versionierung und kontrollierter Veröffentlichung.

Kein vollständiges persönliches Archiv im Internet – sondern eine gezielt kuratierte Knowledge Base.

Obsidian unterstützt die tägliche Arbeit am Wissensbestand. Markdown und das Open Knowledge Format geben den Inhalten eine portable Struktur. Git macht ihre Entwicklung nachvollziehbar. Quartz verwandelt die freigegebene Auswahl in eine navigierbare Website. Eine explizite Allowlist schützt dabei die Grenze zwischen internem Material und öffentlicher Darstellung.