1_Defensive-Publication_Der_Prior-Art_Schutz_NWM_v3.3

Defensive Publication: Datensouveränes Wissensmanagement mit hybrider KI-Architektur in Einzelorganisationen und Netzwerken

Projekt-Code: #NWM (Netz-Werkzeug.Macherei)
Status: Öffentlich – v3.3 (Stable Release)
Datum: 20.05.2026
Verfasser: Netz-Werkzeug.Macherei
Lizenz: CC BY-NC-SA 4.0

Vorgängerversionen:

- v3.0 (15.04.2026): DOI 10.5281/zenodo.19597550 · Wayback-Snapshot 15.04.2026
- v3.2 (22.04.2026): DOI 10.5281/zenodo.19699717 · Wayback-Snapshot HP 22.04.2026 · Wayback PDF 24.04.2026

Verankerung dieser Version (v3.3):

- **Zenodo-DOI:** wird beim Upload vergeben und im begleitenden GitHub-Repository sowie auf der Homepage verlinkt
- **GitHub-Repository:** github.com/Netz-Werkzeug-Macherei/nwm-defpub (Release-Tag: `v3.3`)
- **Homepage:** netz-werkzeug-macherei.de
- **SHA-256 dieser PDF:** wird beim Upload erzeugt und im Repository als `v3.3.sha256` hinterlegt
- **Bitcoin-Blockchain-Verankerung via OpenTimestamps:** siehe begleitende Datei `v3.3.ots` im GitHub-Repository und im Zenodo-Record

> *Hinweis zur Versionierung: v3.1 war ein interner Zwischen-Entwurf und wurde nicht öffentlich publiziert. Die kanonischen veröffentlichten Versionen sind v3.0, v3.2 und die vorliegende v3.3.*

Begriffsverzeichnis

Begriff	Definition
iKi (interne KI)	Lokal ausgeführte KI-Modelle (On-Premise) zur datenschutzkritischen Verarbeitung
cKi (Cloud-KI)	Externe KI-Dienste über API-Gateways für hohe Rechenleistung
HITL (Human-in-the-Loop)	Manuelle, technisch erzwungene Freigabeinstanz vor externer Datenübermittlung; umfasst Anzeige, Bestätigung und Protokollierung
HITL-Logging	Revisionssicheres Append-only Log aller HITL-Entscheidungen zur Auditierbarkeit
RAG (Retrieval-Augmented Generation)	Anreicherung von KI-Anfragen mit kontextuellen Dokumenten aus lokaler Wissensdatenbank
SSOT (Single Source of Truth)	Zentrale, autoritative Datenplattform
SOP (Standard Operating Procedure)	Dokumentierte Standardarbeitsanweisungen für Prozesskonformität
ZDR (Zero-Data-Retention)	Richtlinie zur Nicht-Speicherung von Anfragen durch Gateway-/Inferenz-Anbieter
RBAC (Role-Based Access Control)	Rollenbasierte Zugriffssteuerung für Nutzerberechtigungen
PII (Personally Identifiable Information)	Personenbezogene Daten, die besonderen Schutz erfordern
NER (Named Entity Recognition)	Verfahren zur automatischen Erkennung von Entitäten (Personen, Orte, Organisationen)
DLP (Data Loss Prevention)	Maßnahmen zur Verhinderung unkontrollierten Abflusses sensibler Daten
Föderation	Sichere Verbindung mehrerer unabhängiger Instanzen zu dezentralem Netzwerk
Fine-Tuning	Anpassung vortrainierter KI-Modelle auf domänenspezifische Daten
Ethik-Compliance-Filter	Optional aktivierbare Prüfschicht, die Anfragen vor Cloud-Versand gegen die organisationseigenen, live-bezogenen Ethik-Grundsätze abgleicht und Verstöße manipulationssicher protokolliert
Tamper-Evident	Manipulationsersichtlich (engl.): jede nachträgliche Änderung ist technisch nachweisbar, jedoch nicht zwingend technisch verhindert

Begriff	Definition
OTS (OpenTimestamps)	Offenes Protokoll zur Verankerung von Datei-Hashes in der Bitcoin-Blockchain als manipulationsersichtlicher Zeitstempel

---

1. Technisches Feld

Die vorliegende Offenlegung betrifft ein informationstechnisches System zur Integration von Künstlicher Intelligenz (KI) in die Wissens-Infrastruktur von souveränen Organisationen (KMUs, Vereine, NGOs, Gemeinden, Interessengemeinschaften) sowie datenschutzbewussten Einzelanwendern (Heim-Server-Betreiber).

Insbesondere beschreibt sie eine **auditierbare Orchestrierungsarchitektur mit konfigurierbaren Entscheidungs-Checkpoints und revisionssicherer Protokollierung**, die lokale Datenverarbeitung (On-Premise) mit cloud-basierten KI-Diensten unter Wahrung der Datensouveränität verbindet. Das System umfasst:

- **Hybride KI-Orchestrierung** (Interne iKi + Cloud cKi)
- **RAG-Anbindung** (Retrieval-Augmented Generation) an selbstgehostete Wissensplattformen
- **Human-in-the-Loop** (HITL) als Freigabeinstanz mit konfigurierbarem Trigger-Mechanismus, optional ergänzt um einen **Ethik-Compliance-Filter**
- **Multimodale Verarbeitung** (Text, Audio, Bild, Video) – modular zuschaltbar
- **Agenten-Funktionen** (ausführende KI; Referenz-Implementierungen wie LangChain, AutoGen, CrewAI)
- **Auditierbares Qualitätsmanagement:** Die Architektur ermöglicht prozesskonforme Abläufe durch revisionssicheres HITL-Logging, konsistente Wissensanwendung (SOP-Anbindung via RAG) und auditierbare Datenflüsse. Dies unterstützt Qualitätsmanagement-Anforderungen (z.B. ISO 9001) auf technischer Ebene (ersetzt jedoch keine Zertifizierung).
- **Dezentrale Netzwerkstruktur:** Das System unterstützt verteilte Zugriffsszenarien über verschlüsselte VPN-Verbindungen, wodurch mehrere Standorte sicher miteinander verbunden werden können, ohne dass die Infrastruktur im öffentlichen Internet erreichbar sein muss.

Zweck dieser Veröffentlichung: Sicherung des Standes der Technik/Wissenschaft (Prior Art) gemäß §3 PatG (DE) bzw. Art. 54 EPÜ zur Verhinderung von Patentansprüchen Dritter auf das beschriebene Systemdesign.

Bezug zu Vorgängerversionen: Diese Veröffentlichung baut auf der Defensive Publication NWM v3.0 vom 15.04.2026 (DOI 10.5281/zenodo.19597550) und v3.2 vom 22.04.2026 (DOI 10.5281/zenodo.19699717) auf. Die Kernarchitektur (hybride iKi/cKi-Orchestrierung mit Routing-Logik, HITL, RAG, Multi-Level-Guardrails) ist seit dem 15.04.2026 als Prior Art

etabliert. Die vorliegende Version v3.3 erweitert und präzisiert die technische Beschreibung, ergänzt den optionalen Ethik-Compliance-Filter sowie technische Maßnahmen zur Admin-Resistenz, grenzt sie zu zwischenzeitlich publizierter, ähnlich gerichteter Literatur ab und konsolidiert Implementierungs-Spezifikationen.

Hinweise zu Produkten und Aktualität:

- Alle im Dokument genannten Produkte, Dienste und Technologien (z.B. Nextcloud, Ollama, OpenRouter) dienen als Referenz-Implementierungen. Das System ist technologie-agnostisch und übertragbar auf vergleichbare Lösungen mit ähnlichen Spezifikationen.
- Die Einzelkomponenten sind nach Vorlieben einsetzbar.
- Die Entwicklung im Bereich der Künstlichen Intelligenz schreitet mit hoher Geschwindigkeit voran. Produktbezeichnungen, Modellversionen und technische Spezifikationen können bereits wenige Wochen nach Veröffentlichungsdatum veraltet sein.

---

2. Hintergrund & Problemstellung

2.1 Technisches Problem

Bestehende Lösungen zur KI-Integration neigen dazu, entweder vollständig cloud-basiert zu sein (mit Risiken hinsichtlich Datenschutz, Datenabfluss und Vendor-Lock-in) oder vollständig lokal zu operieren (mit Einschränkungen hinsichtlich Rechenleistung und Modellkapazität).

Für Organisationen und Einzelanwender mit hohen Datenschutzanforderungen (z.B. aufgrund von DSGVO, Berufsgeheimnissen oder persönlicher Präferenz) besteht ein technisches Problem darin, leistungsfähige KI-Modelle zu nutzen, ohne sensible Daten unkontrolliert an externe Anbieter zu übermitteln.

Ein zusätzlicher operativer Nebenvorteil der hier beschriebenen Architektur: Auf der lokalen Plattform und in lokal verarbeiteten Anfragen können Klarnamen, reale Bezeichnungen und sensitive Bezüge unverstellt verwendet werden. Erst beim Cloud-Versand greift die Anonymisierungs-Pipeline. Dies vereinfacht die interne Wissensarbeit erheblich, da kein vorgelagertes manuelles Verklausulieren in Arbeitsdokumenten mehr nötig ist.

2.2 Zielgruppen & Anforderungsprofil

Zielgruppe	Bisherige Lösungen	Architektureller Ansatz
Enterprise	Kong AI Gateway, MuleSoft (teuer, komplex)	Nicht primärer Fokus

Zielgruppe	Bisherige Lösungen	Architektureller Ansatz
KMU (<500 MA)	Keine spezifischen integrierten Lösungen	**Fokus: Hybride Orchestrierung**
Einzelanwender	PrivateGPT, Ollama (lokal, kein Team)	**Fokus: Team-Funktion & Sovereignty**

Technischer Kern: Es geht nicht nur um KI. Es geht um **eigenes Wissen** (Termine, Kontakte, Projekte, Dateien, SOPs). Sicher gespeichert. KI-gestützt durchsuch- und bearbeitbar. Im Team nutzbar, ohne dass externe Dritte mitlesen.

---

3. Charakteristische Merkmalskombinationen und offenbarte Ausführungsformen

Die nachfolgend beschriebene Architektur zeichnet sich durch die Kombination folgender Merkmale aus, die in Sequenz und Wechselwirkung das Schutzziel der Datensouveränität bei gleichzeitiger Nutzung externer Hochleistungs-KI sicherstellen. **Eine bevorzugte Ausführungsform** umfasst die Merkmale M1–M5 in der beschriebenen Anordnung; weitere offenbarte Ausführungsformen umfassen Teilkombinationen, funktional äquivalente Varianten und alternative Sequenzen. Diese Defensive Publication offenbart **sowohl die bevorzugte Ausführungsform als auch alle hierin beschriebenen Varianten** als Stand der Technik.

M1: Lokale Verarbeitungsebene (iKi)
Eine lokale Verarbeitungsebene (iKi) mit mindestens einem Open-Weight-Sprachmodell, die alle eingehenden Nutzeranfragen mindestens **einer lokalen Sensibilitäts-Klassifikation** (Confidence-Score-basiert; Default-Schwellwert siehe Abschnitt 6.1) unterzieht, bevor eine etwaige Cloud-Übermittlung initiiert wird. Funktional äquivalent gelten:

- Open-Weight-Sprachmodelle ab ca. 7 Milliarden dense Parametern (Referenz: 8B–13B Modelle wie Llama-3.1+, Mistral, Qwen)
- Mixture-of-Experts-Modelle ab ca. 30B Parametern mit ≥3B aktiven Parametern
- Klassische ML-Klassifikatoren oder regelbasierte Erkennungssysteme mit vergleichbarer operativer Eignung für Sensibilitäts-Klassifikation und PII-Erkennung im beschriebenen Workflow

Notfall-Ausnahmebetrieb: Bei Hardware-Defekt oder geplanter Wartung der iKi-Ebene kann das System in einen *degradierten Notbetriebs-Modus* übergehen, in dem Anfragen entweder abgewiesen werden (Fail-Safe-Default) oder mit expliziter, geloggter Nutzer-Bestätigung an cKi weitergeleitet werden. Dieser Notbetriebs-Modus ist **kein regulärer Betriebspfad** und muss durch Hardware-Watchdog oder vergleichbare technische Auslöser (z.B. mehrfacher iKi-Health-Check-Fehler) aktiviert werden, nicht durch bloße Konfigurationswahl.

M2: Mehrstufige PII-Filter-Pipeline
Eine mehrstufige PII-Filter-Pipeline mit **mindestens drei**, in der Referenz-Implementierung vier Stufen:

- **(a)** Regelbasierte deterministische Erkennung (NER + Regex) als Basisebene für strukturierte PII-Muster
- **(b)** LLM-basierte semantische Umformulierung als ergänzende Sicherungsebene; verändert Beziehungsstrukturen und Kontext-Hinweise, **erhält jedoch die typisierten Platzhalter-Token aus Stufe (a)** zur Wahrung der Reversibilität
- **(c)** Reversibles Mapping in flüchtigem, nicht-persistentem Speicher (RAM-only, session-scoped)
- **(d)** Verifikationsstufe (leichtgewichtig, NER-Cross-Check und Regex-Recheck)

Stufe (b) ist verpflichtender Bestandteil der bevorzugten Ausführungsform und wird ausgelöst, sobald ein PII-Verdacht der Stufe (a) oder eine Cloud-Versand-Vorbereitung vorliegt (siehe Abschnitt 6.1).

M3: Konfigurierbares HITL-Gate mit optionalem Ethik-Compliance-Filter
Ein konfigurierbares HITL-Gate, das zwischen iKi-Vorverarbeitung und cKi-Übermittlung als Checkpoint operiert, optional ergänzt um einen aktivierbaren Ethik-Compliance-Filter mit Live-Bezug auf organisationseigene Ethik-Grundsätze. Das Gate ist **nicht-umgehbar für jede zur cKi-Übermittlung vorgesehene Anfrage**, sofern keine vorherige, im Audit-Log protokollierte Vorautorisierung des Nutzers vorliegt. Vorautorisierungs-Modi (Session, Pattern) sind explizit vom Nutzer zu aktivieren und bleiben jederzeit revidierbar (siehe Abschnitt 6.2).

M4: RBAC-bewusste RAG-Anbindung
Eine RAG-Anbindung an eine selbstgehostete SSOT-Plattform mit **RBAC-Berücksichtigung im Retrieval-Prozess**. Diese Defensive Publication offenbart zwei Ausführungsformen:

- **Vorfilterung:** RBAC-Metadatenfilter direkt in der Vektor-DB-Abfrage (bevorzugt für Performance bei großen Beständen)
- **Nachfilterung:** RBAC-Filterung der Top-k retrieved Chunks vor Kontext-Injection (einfacher zu implementieren, geringfügig höhere Latenz)

In beiden Varianten erfolgt zusätzlich eine PII-Filterung der retrieved Chunks vor Kontext-Injection (siehe Abschnitt 4.3).

M5: Multi-Provider-Routing-Schicht
Eine Multi-Provider-Routing-Schicht mit **mindestens zwei unabhängigen Cloud-Anbietern** zur Reduktion von Anbieter-Abhängigkeit (Anti-Vendor-Lock-in); hilfsweise oder ergänzend regionale Routing-Profile innerhalb eines Anbieters, sofern dies eine Korrelation einzelner Anfragen zu einem Nutzerprofil statistisch erschwert. Die Routing-Schicht arbeitet

mit ZDR-Policy-Filterung und konfigurierbaren Routing-Regeln (z.B. regionale Verteilung, Sensibilitäts-basierte Auswahl, Kostenoptimierung) (siehe Abschnitte 6.3 und 6.4).

M6: Optionale Föderationsschicht
Eine optionale Föderationsschicht für sichere Inter-Instanz-Kommunikation auf Applikationsebene, abzugrenzen vom Standard-VPN-Zugriff einzelner Nutzer. Die Föderation erlaubt den vernetzten Austausch zwischen mehreren NWM-Instanzen – mit oder ohne eigene iKi-Ressourcen pro Instanz – über verschlüsselte Kanäle ohne öffentlichen Internet-Zugang (siehe Abschnitt 6.5).

M7: Optionale Multimodal-Erweiterung
Eine optionale Multimodal-Erweiterung, gegliedert in:

- **M7a:** Audio-Verarbeitung – stark empfohlen lokal zur Wahrung biometrischer Stimm-Daten
- **M7b:** Bildanalyse – lokal mit erhöhten VRAM-Anforderungen
- **M7c:** Bildgenerierung – lokal nur mit deutlich erhöhter Hardware-Ausstattung praktikabel
- **M7d:** Videoverarbeitung – Cloud bevorzugt

Alle M7-Submerkmale operieren unter Beibehaltung der M1-M5-Sicherheitskette.

Gesamtbild:
Die Merkmale **M1–M5 bilden die bevorzugte Ausführungsform** und werden in dieser Kombination als Stand der Technik offenbart. **M6 und M7 sind optionale Erweiterungen.** Der Ethik-Compliance-Filter innerhalb von M3 ist optional aktivierbar. Diese Defensive Publication erfasst sowohl die vollständige Kombination M1–M5 als auch **explizit Teilkombinationen, alternative Sequenzen und funktional äquivalente Varianten der einzelnen Merkmale**, soweit sie in dieser Schrift beschrieben sind. Einzelne Merkmale für sich betrachtet können separaten, vorbestehenden Stand der Technik darstellen; offengelegt wird hier insbesondere ihre konkrete Verkettung und die offenbarten Varianten.

---

4. Systemarchitektur

4.1 Zentrale Wissensplattform (Single Source of Truth)

Als zentrale Ablage- und Kollaborationsplattform dient eine selbstgehostete oder Managed-Hosting-Cloud-Instanz (beispielsweise basierend auf Nextcloud-Technologie).

Funktionen:

- Dokumentenablage (Dateien, PDFs, Bilder, Videos)
- Kalender, Kontakte, Aufgaben (Kanban-Boards, z.B. Deck)
- Wiki-Funktionalität (Markdown-basiert)
- Kommunikation (z.B. Nextcloud Talk)

- **SSOT:** Single Source of Truth – alle Daten an einem Ort

Strukturierung:

- **Geteiltes Wissen:** Team-Zugriff über zentrale Plattform
- **Persönliches Wissen:** Internes Denken, Entwurf, persönliche Notizen (z.B. synchronisierte Vaults)

Hosting-Optionen:

- Hosting-Optionen umfassen Managed Hosting (geringerer Setup-Aufwand) oder eigener Server (Proxmox/Linux) für volle Kontrolle. Beide Pfade sind unterstützt.
- **Cloud-agnostisch:** Nextcloud ist Referenz, aber übertragbar auf OwnCloud, Seafile, etc.

Netzwerk-Zugriff:

Das System unterstützt verteilte Zugriffsszenarien: Einzelne Nutzer können über verschlüsselte VPN-Verbindungen (z.B. Tailscale, WireGuard) auf die Plattform zugreifen, ohne dass diese im öffentlichen Internet erreichbar sein muss. Die Architektur erlaubt zudem die Trennung von Plattform (Speicher) und KI-Ressourcen (Rechenleistung), die sicher über das interne Netzwerk verbunden werden. Die Inter-Instanz-Kommunikation auf Applikationsebene ist als optionale Föderationsschicht (M6) separat geregelt (siehe Abschnitt 6.5).

4.2 Hybrider KI-Orchestrator

Ein zentrales Merkmal des Systems ist eine Orchestrierungsschicht, die eingehende Anfragen analysiert und routet.

Sequenzieller Datenfluss (bevorzugte Ausführungsform):

```
Eingabe
  → iKi (Sensibilitäts-Klassifikation + RAG-Retrieval + adaptive PII-
Pipeline)
  → HITL-Gate (für jede zur Cloud vorgesehene Übermittlung;
               bei rein lokaler Verarbeitung entfällt das Gate)
  → cKi (Cloud-Verarbeitung, falls erforderlich)
  → iKi (Re-Identifikation, falls Anonymisierung erfolgte; Formatierung)
  → Ausgabe an Nutzer
```

Sequenz innerhalb der iKi-Verarbeitung (Klarstellung):

Die iKi-Stufe verarbeitet Nutzer-Input und RAG-Kontext in folgender Reihenfolge: (1) Sensibilitäts-Klassifikation des User-Inputs, (2) RAG-Retrieval mit RBAC-Vorfilterung gemäß Abschnitt 4.3, (3) Zusammenführung von User-Input und retrieved Chunks, (4) gemeinsame Verarbeitung durch die mehrstufige PII-Pipeline gemäß Abschnitt 6.1, (5) Übergabe an HITL-

Gate. Damit ist sichergestellt, dass auch in retrieved Chunks enthaltene PII vor jedem Cloud-Versand erfasst und behandelt wird.

Adaptive Pipeline-Aktivierung:

Um Praktikabilität auf Consumer-Hardware sicherzustellen, ist die Pipeline in zwei klar abgegrenzten Verarbeitungspfaden organisiert:

Lane	Bedingung	Verarbeitungspfad	HITL-Gate erforderlich?	Typische Latenz
Local-Lane	Anfrage rein lokal verarbeitbar; **kein Cloud-Versand vorgesehen**	iKi-Klassifikation → iKi-Antwort	Entfällt – kein Cloud-Versand	~2–5 Sek.
Cloud-Lane	Cloud-Versand vorgesehen (durch Nutzer-Wahl oder Routing-Logik)	iKi-Klassifikation → vollständige PII-Pipeline → HITL-Gate → cKi → Re-Identifikation	Ja, für jede Anfrage; Vorautorisierung möglich	~10–120 Sek. (System-Verarbeitungszeit ohne Nutzer-Bedienzeit)

Die Latenz-Angaben sind Größenordnungen und beziehen sich auf reine **System-Verarbeitungszeit**; HITL-Bedienzeit (Nutzer-Sichtprüfung) ist nutzerabhängig und nicht eingerechnet. Konkrete Werte hängen von Hardware, Modellgröße, Kontext-/Prompt-Länge, RAG-Volumen und Cloud-Latenz ab.

Wichtig: Die Architektur sieht **keine Lane vor, in der Cloud-Versand ohne HITL-Bestätigung oder ohne vorherige protokollierte Vorautorisierung erfolgt**. Damit ist M3 ("non-bypassable für jede Cloud-Übermittlung") konsistent durchgehalten.

Gateway-agnostisch: Das System nutzt referenziell Multi-Model-Gateways (z.B. OpenRouter), ist aber nicht darauf beschränkt.

4.2.1 Lokale Verarbeitungsebene (iKi – interne KI)

Aspekt	Spezifikation (Beispiele)
Software	Ollama, LM Studio, Msty Studio (Referenz-Implementierungen)
Modelle	typischerweise 8B–13B Parameter (Llama-3.1+, Mistral, Qwen oder Nachfolgegenerationen) oder funktional äquivalent gemäß M1
Hardware	GPU ≥16 GB VRAM (z.B. RTX 4060 Ti 16GB) als Basis-Empfehlung für Text-only; für Multimodal-Submerkmale erhöhte Anforderungen siehe Abschnitt 4.4 und 5.1
Einsatz	Sensibilitäts-Klassifikation aller Anfragen, vollständige PII-Anonymisierung sensibler Anfragen, RAG-Retrieval, Re-Identifikation der cKi-Antworten

4.2.2 Cloud-Verarbeitungsebene (cKi – cloudbasierte KI)

Aspekt	Spezifikation (Beispiele)
Gateway	Multi-Model-Gateway-Anbieter (Referenz: OpenRouter); direkt-API möglich
Modelle	typischerweise ≥70B Parameter oder Frontier-Modellklasse
Datenschutz	EU-Only wo möglich, ZDR (Zero-Data-Retention), zusätzliche Filter auf Gateway-Ebene
Einsatz	Komplexe Aufgaben, hohe Leistung, wenn iKi nicht ausreicht

4.2.3 Routing-Logik (Orchestrator-gesteuert)

Die Routing-Entscheidung erfolgt anhand eines zweidimensionalen Klassifikationsschemas (Sensibilität: `+` = sensibel, `–` = nicht sensibel; Leistung: `+` = hoher Bedarf, `–` = niedriger Bedarf):

Fall	Sensibilität	Leistung	Lösung
A	–	–	→ iKi (Local-Lane bevorzugt) oder cKi nach Nutzer-Wahl mit HITL
B	–	+	→ cKi mit HITL
C	+	–	→ iKi (Local-Lane)
D	+	+	→ L1 / L2 / L3 (siehe unten)

Lösungsoptionen für Fall D (Sensibel UND Hochleistung):

- **L1:** Lokale Anonymisierung (iKi) → dann Cloud (cKi) mit HITL; **Standard-Fall**
- **L2:** In mehrere Fragen aufteilen, nur über iKi (Verzicht auf Hochleistung)
- **L3:** Direkt mit cKi nach bewusster Nutzer-Entscheidung und HITL-Bestätigung (Akzeptanz des reduzierten, aber nicht vollständig eliminierbaren Restrisikos)

In allen Fällen mit Cloud-Versand greift das HITL-Gate (M3). Die Local-Lane (Fälle A Local-Lane, C) erfordert kein HITL, weil keine externe Datenübermittlung erfolgt.

Sensibilitäts-Klassifikation: Erfolgt durch lokalen PII-Detector (Confidence-Score, siehe Abschnitt 6.1). Liegt der Score über einem konfigurierbaren Schwellwert (Default: **0.7**), wird die Anfrage als sensibel klassifiziert.

Leistungs-Klassifikation: Wird durch Komplexitäts-Heuristiken (Token-Länge, semantische Komplexität, Aufgabentyp) abgeschätzt; konfigurationsspezifisch durch den Implementierer anpassbar.

4.3 RAG-Implementierung (Retrieval-Augmented Generation)

Komponente	Technologie (Beispiele)	Funktion
Vektor-Datenbank	ChromaDB, Qdrant, LlamaIndex	Speichert Dokumenten-Embeddings
Indexierung	Plattform-Inhalte (PDF, MD, DOCX, etc.)	Automatische Extraktion bei Upload
Retrieval	iKi fragt Vektor-DB vor Generierung der cKi-Anfrage	Kontext wird lokal angereichert
Orchestrierung	Python/FastAPI + litellm, n8n, Node-RED	Workflow-Automatisierung

Spezifischer Datenfluss:

1. Dokument-Upload → automatisches Chunking (Default: ca. 800–1200 Token pro Chunk, Overlap ca. 100 Token; konfigurierbar)
2. Embedding-Generierung **lokal** via iKi (z.B. nomic-embed-text, bge-m3) – KEINE Cloud-Embeddings
3. Speicherung in Vektor-DB mit Metadaten (Quelle, RBAC-Tag, Sensitivitäts-Tag, Zeitstempel)
4. Bei Query: lokale Embedding-Generierung → Top-k Retrieval (Default: **k=8**, Cosine-Similarity-Threshold: **0.7**; konfigurierbar)
5. **RBAC-Berücksichtigung im Retrieval-Prozess** als sicherheitsrelevantes Merkmal (M4):
   - **Bevorzugte Variante:** Vorfilterung im Vektor-DB-Query mittels RBAC-Metadatenfilter (Performance-optimal)
   - **Alternative Variante:** Nachgelagerte RBAC-Filterung der Top-k-Ergebnisse vor Kontext-Injection (einfacher zu implementieren)
6. PII-Filter prüft auch retrieved Chunks (nicht nur User-Input); User-Input und RAG-Kontext werden gemeinsam anonymisiert
7. Kontext-Injection in Prompt-Template; HITL-Gate erhält die anonymisierte Gesamt-Anfrage zur Sichtprüfung

Hinweis zu Default-Werten: Die in dieser Schrift genannten Schwellwerte sind Beispiel-Defaults aus der Referenz-Implementierung. Sie sind konfigurierbar; aktualisierte Werte und Begründungen werden im begleitenden GitHub-Repository fortlaufend bereitgestellt.

Hinweis zur Performance: Die RBAC-Vorfilterung in der Vektor-DB-Abfrage ist Performance-optimal; die nachgelagerte Filterung kann bei sehr großen Dokumentenbeständen zu erhöhter Latenz führen. Optimierungsstrategien werden im begleitenden GitHub-Repository fortlaufend ergänzt.

4.4 Multimodale Architektur (M7-Detaillierung)

Das **grundsätzlich textbasierte System** ist erweiterbar auf multimodale Verarbeitung. Die folgende Tabelle zeigt die Hardware-Anforderungen und empfohlene Verarbeitungsorte:

Submerkmal	Modalität	Lokal (iKi)	Cloud (cKi)	Anmerkung
M7a	Audio (Transkription, Speech-to-Text)	Whisper-Modelle, ab ~4-8 GB VRAM	Whisper API, ElevenLabs etc.	**Lokal stark empfohlen** zur Wahrung biometrischer Stimm-Daten
M7b	Bildanalyse (Vision-Language)	LLaVA, Llama-3.2-Vision oder Nachfolger (ab ~16 GB VRAM bei Solo-Betrieb; ≥24 GB VRAM bei Parallelbetrieb mit Sprachmodell)	GPT-4V, Claude Vision	Lokal mit 16 GB VRAM machbar nur bei Modell-Swap; ≥24 GB VRAM für Parallelbetrieb
M7c	Bildgenerierung	Stable Diffusion (empfohlen ≥24 GB VRAM mit Sprachmodell-Swap; ≥48 GB VRAM für reibungslosen Parallelbetrieb)	DALL-E 3, Midjourney	Cloud meist effizienter
M7d	Video-Verarbeitung	Frame-Extraktion + Whisper (begrenzt)	Sora, Runway	Lokal für Volltext-Video kaum praktikabel

Text-Verarbeitung gilt als Basis-Modalität und ist nicht Teil von M7.

Die PII-Filterung wird modalitätsspezifisch erweitert (z.B. Gesichts-Anonymisierung in Bildern, Stimm-Anonymisierung optional bei Audio-Cloud-Versand).

Die Sicherheitskette M1–M5 bleibt für alle Modalitäten erhalten.

4.5 Agenten-Funktionen

Das System ist kompatibel mit ausführenden KI-Agenten:

Agenten-Framework	Zweck	Integration
LangChain	Workflow-Framework	Orchestrator-Integration
AutoGen	Multi-Agenten-Systeme	Für komplexe Aufgaben

Agenten-Framework	Zweck	Integration
CrewAI	Rollen-basierte Agenten-Teams	Für Team-Automatisierung
Nextcloud LLM2 / Context Chat	Offizielle Nextcloud-KI-Apps	Basis-Integration

Web-Tools als Erweiterungs-Funktionen: Der Orchestrator unterstützt optional die Anbindung externer Web-Tools, insbesondere:

- **Web-Search:** Anbindung an Suchmaschinen-APIs (z.B. Brave Search API, DuckDuckGo, Kagi, Tavily) zur Bereitstellung aktueller Information für die KI-Verarbeitung. Suchanfragen können in der bevorzugten Ausführungsform vor Versand durch die mehrstufige PII-Pipeline gemäß Abschnitt 6.1 anonymisiert werden, sofern sie nutzerbezogene Bestandteile enthalten.
- **Web-Fetch:** Gezieltes Abrufen einer bekannten URL und Extraktion des Inhalts (HTML, PDF, Markdown) zur Aufnahme in den Kontext. Beim Fetchen externer Inhalte erfolgt eine Sensibilitäts-Klassifikation der zurückgelieferten Inhalte gemäß M1, bevor diese in nachfolgende Cloud-Anfragen einfließen.
- **Datums- und Zeitabfragen:** Lokale oder externe Bereitstellung aktueller Zeitstempel und Datum-Bezüge für die KI-Verarbeitung, ohne nutzerbezogene Daten an externe Dienste zu übermitteln.

Diese Web-Tools operieren **innerhalb** der bestehenden Sicherheitskette M1–M5: Aus den Tools resultierende Cloud-Anfragen unterliegen denselben HITL-Anforderungen wie direkte Nutzeranfragen; aus Tools zurückkehrende Inhalte werden vor Aufnahme in den Kontext klassifiziert. Die hierfür erforderliche, im Audit-Log protokollierte Vorautorisierung folgt der Pattern- oder Session-Logik gemäß Abschnitt 6.2.

Hinweis zur HITL-Konsistenz: Die Integration in den Orchestrator erfolgt unter **differenzierter Anwendung** der M1–M5-Sicherheitskette. Direkte Nutzeranfragen mit Cloud-Versand durchlaufen in jedem Fall das HITL-Gate gemäß Abschnitt 6.2 (siehe M3). Für agenten-getriebene interne Aktionen (Dateilesen, lokale DB-Zugriffe, lokale iKi-Calls ohne Cloud-Versand) entfällt das HITL-Gate analog zur Local-Lane. **Externe Agent-Calls** (Cloud-Versand durch Agenten) erfordern jedoch eine **explizite, zeitlich befristete und im Audit-Log protokollierte Vorautorisierung** durch den Nutzer (Pattern- oder Session-Vorautorisierung gemäß 6.2). Damit ist auch agenten-getriebener Cloud-Versand niemals "unbeobachtet".

---

5. Hardware- & Software-Referenz

5.1 Hardware-Klassifikation

Kategorie	Beispiele	Einsatz	Kosten (ca.) (Stand Q2/2026)	Empfehlung
Consumer GPU	RTX 4060 Ti 16GB	Lokale Text-Inferenz (8B–13B), M7a (Audio); M7b/M7c nur sequenziell mit Modell-Swap (Swap-Latenz 5–30 s)	500–700€	**Text-only Standard**
Mid-Range GPU	RTX 4080/4090 16-24GB	Text + M7b (Bildanalyse)	1.000–2.000€	**Multimodal Bildanalyse**
Enthusiast GPU	RTX 4090 24GB	Text + M7b parallel oder Text + M7c sequenziell mit Modell-Swap; reibungsloser Parallelbetrieb aller Modalitäten erfordert ≥48 GB	1.500–2.000€	**Multimodal kombiniert (mit Modell-Swap)**
Profi/Multi-GPU	2× RTX 4090, A6000 48GB	Voll-Multimodal lokal (M7a–M7d) reibungslos parallel	3.500–8.000€	**Voll-Multimodal lokal**
Datacenter GPU	A100 40/80GB, H100	Training, große Inferenz	10.000–30.000€	Cloud mieten empfohlen
Cloud-GPU	RunPod, Vast.ai	Fine-Tuning, Training	1–5€/Stunde	Für Fine-Tuning-Szenarien
Mac Studio	M2/M3 Ultra (≥64 GB Unified Memory)	Lokale KI (macOS), Text + leichtes Multimodal	4.000–8.000€	macOS-Option
Intel NUC	Mini-PC (CPU-Inferenz)	Kompakte iKi-Hosting (langsam)	800–1.500€	KMU-Nischenlösung

Klare Hardware-Staffelung nach Nutzungsprofil:

- **Text-only (Standard-KMU/Verein):** ≥16 GB VRAM
- **Text + Bildanalyse (M7b):** ≥24 GB VRAM empfohlen
- **Text + Bildgenerierung (M7c):** ≥24 GB VRAM mit Modell-Swapping; ≥48 GB für Parallelbetrieb
- **Voll-Multimodal lokal (M7a–M7d):** ≥48 GB VRAM oder Mehrkarten-Setup; alternativ Cloud-Bursting für M7c/M7d

Hinweis zu Betriebskosten: Neben der Hardware-Anschaffung sind Stromkosten zu berücksichtigen. Je nach GPU, Idle-/Lastprofil und Strompreis können diese vom niedrigen zweistelligen bis in den dreistelligen Eurobereich pro Monat reichen.

Fazit: Für KMU-Infrastrukturen ergeben sich hardwareseitige Grenzen für lokale Inferenz. Hochperformante Cloud-KI (Frontier-Modelle) erfordert weiterhin Cloud-Nutzung. 16 GB VRAM ermöglichen lokale Text-Inferenz für Modelle im Bereich 8B–13B Parameter, was für Standardaufgaben ausreichend ist.

5.2 Deployment-Szenario

Das System ist als **mehrstufiges Deployment** implementierbar. Realistische Aufwandsabschätzung:

Reifegrad	Inhalt	Zeitaufwand (qualifizierter Admin)
Proof-of-Concept	iKi + lokale Wissens-Plattform, ohne RAG, ohne HITL-UI	1–2 Tage
Pilotinstallation	Vollständige Pipeline iKi + RAG + HITL-Gate-Basis-UI, Single-User	1–2 Wochen
Produktionsreife KMU-Installation	Multi-User, RBAC, Audit-Logging, Föderation, Security-Review	4–12 Wochen

Schritte einer Standard-Installation (Pilot):

1. Setup-Skript auf Server/PC ausführen (Linux/Proxmox-Basis)
2. Plattform-Instanz freigeben (Server-Speicherplatz, Verzeichnisstruktur)
3. KI-Orchestrator + vorkonfigurierte Prompts/Skripte installieren
4. Internes Modell herunterladen (z.B. Llama-3.1+ oder Nachfolger)
5. Interne Dateien einbeziehen und indexieren lassen (RAG-Initialisierung)
6. RAG-Parameter konfigurieren (Default-Werte: siehe Abschnitt 4.3)
7. Weitere Mitglieder einbinden (Berechtigungen via RBAC)
8. Auf weiteren Rechnern installieren (Sync-Clients)

Hinweis: Konkrete Implementierungs-Skripte und Konfigurationsbeispiele werden im begleitenden GitHub-Repository (github.com/Netz-Werkzeug-Macherei/nwm-defpub) fortlaufend bereitgestellt. Der konkrete Aufwand hängt von der individuellen Infrastruktur und der gewählten Komponenten-Kombination ab; die obenstehenden Reifegrad-Angaben gelten für qualifiziertes Personal.

---

6. Verfahren zur Datensicherheit (Defense-in-Depth)

6.1 Lokale Vorfilterung (iKi-Ebene) – Mehrstufige Pipeline

Bevor sensible Daten das lokale Netzwerk verlassen, durchlaufen sie eine interne KI-Schicht (iKi) mit folgender mehrstufiger Pipeline (mindestens drei, in der Referenz-Implementierung vier Stufen). Die Pipeline läuft **vollständig lokal vor dem HITL-Gate** und damit vor jedem zur Cloud vorgesehenen Versand.

Übersicht – wann was passiert:

Verarbeitungsschritt	Wann	Wo
iKi-Sensibilitäts-Klassifikation (M1)	bei jeder Anfrage, sofern iKi betriebsbereit	Lokal
Vollständige PII-Pipeline (M2, alle Stufen)	wenn PII detektiert ODER Cloud-Versand vorgesehen	Lokal
HITL-Gate (M3)	bei Cloud-Versand (in jedem Fall, sofern keine Vorautorisierung), oder bei Trigger gem. 6.2	Lokal vor Cloud
Cloud-Versand	nur nach erfolgter HITL-Bestätigung oder vorab erteilter, geloggter Vorautorisierung	Cloud
Re-Identifikation	nach Cloud-Antwort, falls vorher anonymisiert	Lokal

Pipeline-Ablauf (Referenz-Implementierung):

- **Eingabe** durch Nutzer
- → **Stufe 1:** Regelbasierte Erkennung (NER/Regex) — *deterministische Basisebene*
- → **Stufe 2:** LLM-basierte semantische Umformulierung — *erhält die Platzhalter aus Stufe 1, formuliert Beziehungsstrukturen um*
- → **Stufe 3:** Reversibles Mapping (RAM-only) — *speichert Klartext↔Platzhalter-Zuordnung für die spätere Re-Identifikation der cKi-Antwort*
- → **Stufe 4:** Verifikation (leichtgewichtig) — *Cross-Check vor HITL-Übergabe*
- → **HITL-Gate** (Abschnitt 6.2)
- → **ggf. cKi** (Cloud-Versand)
- → **Antwort der cKi** (mit ggf. erhaltenen Platzhaltern)
- → **Re-Identifikation** via Mapping aus Stufe 3 (Token-Lookup; Fallback bei beschädigten Platzhaltern: semantisches Matching mit HITL-Eskalation)
- → **Ausgabe** an Nutzer

Stufe 1 – Regelbasierte deterministische Erkennung (NER/Regex):

- Erkennung definierter strukturierter Muster (E-Mail-Adressen, Telefonnummern, IBAN, Postadressen, Personennamen via NER-Modell – z.B. spaCy `de_core_news_lg` für

deutschsprachige Inhalte, analog mehrsprachige Modelle wie `xlm-roberta-base-ner-hrl` für andere Sprachen) und regulärer Ausdrücke
- Ersetzung durch typisierte Platzhalter-Token: `[PERSON_1]`, `[ORG_1]`, `[LOC_1]`, `[EMAIL_1]`, etc.
- **Charakter:** Strukturierte, definierten Mustern entsprechende PII werden mit hoher Zuverlässigkeit erkannt; falsch-positive Erkennungen kommen vor und sind Teil der typischen NER/Regex-Charakteristik. **Keine Vollständigkeitsgarantie für sämtliche denkbaren PII-Formen** (z.B. seltene Schreibweisen, indirekte Identifikatoren).

Stufe 2 – LLM-basierte semantische Umformulierung:

- Die iKi (lokales Modell, gemäß M1) erhält die in Stufe 1 vorverarbeitete Anfrage mit System-Prompt zur Abstrahierung von Beziehungsstrukturen und Kontext-Hinweisen
- **Wichtig:** Die typisierten Platzhalter aus Stufe 1 (`[PERSON_1]` etc.) werden **erhalten**; nur Beziehungsstrukturen und Kontextbezüge werden umformuliert. Damit bleibt das Mapping aus Stufe 3 reversibel.
- Beispiel-Transformation:
  - Input nach Stufe 1: *"`[PERSON_1]` (Schwester der anfragenden Person) hatte folgenden Traum..."*
  - Output Stufe 2: *"`[PERSON_1]` aus dem persönlichen Umfeld der anfragenden Person berichtete folgenden Traum..."*
- **Charakter:** Probabilistisch (LLM-basiert), kann semantische Bezüge erkennen, die Stufe 1 entgehen, bietet aber keine 100%-Garantie. Fungiert als ergänzende Sicherungsebene.

Stufe 3 – Reversibles Mapping (Session-scoped, RAM-only):

- Platzhalter-Klartext-Zuordnungen werden in lokaler Lookup-Tabelle (RAM-only, session-scoped) gespeichert
- Nach cKi-Antwort erfolgt Re-Identifikation durch Rückersetzung der Platzhalter
- Lookup-Tabelle wird nach Session-Ende gelöscht (kein persistentes Speichern)
- Bei längeren Sessions sind Speicher-Schutzmaßnahmen (z.B. mlock, Memory-Encryption) optional empfohlen
- **Hinweis zum Crash-Verhalten:** Geht der iKi-Prozess zwischen Cloud-Versand und Re-Identifikation verloren (z.B. Prozess-Crash, Stromausfall), kann die zurückkehrende cKi-Antwort nicht mehr re-identifiziert werden und besteht ggf. nur noch aus Platzhaltern. Dies ist sicherheitstechnisch korrekt (keine persistenten Klartext-Mappings), kann aber zu Bedienungsverlust führen. Aus QM-Sicht wird ein Audit-Log-Eintrag *"Cloud-Versand erfolgte, Re-Identifikation gescheitert"* erzeugt.

Stufe 4 – Verifikation:

- Sekundärer Verifikationsschritt mittels NER-Cross-Check (anderes Modell oder andere Konfiguration als Stufe 1) und Regex-Recheck der **bereits anonymisierten Anfrage**

- **Diese Verifikation wird ausgelöst, wenn die Pipeline einen Cloud-Versand vorbereitet, und läuft als finaler Cross-Check vor dem HITL-Gate.**
- Bei Detektion von Reststellen erfolgt entweder eine erneute Pipeline-Iteration ab Stufe 1 (bei niedrigem Schweregrad) oder eine Eskalation an das HITL-Gate (bei höherem Schweregrad oder wiederholten Reststellen).

Konfigurierbare Schwellwerte und Beispiel-Defaults: Folgende Default-Werte gelten in der Referenz-Implementierung; sie sind sämtlich konfigurierbar und werden im begleitenden GitHub-Repository fortlaufend dokumentiert und ggf. an aktuelle Modell-Generationen angepasst:

Parameter	Beispiel-Default	Hinweis
PII-Confidence-Schwellwert	**0.7**	Score über diesem Wert klassifiziert eine Anfrage als sensibel
RAG Top-k	**8**	Anzahl der retrieved Chunks pro Query
RAG Similarity-Threshold	**0.7**	Cosine-Similarity-Mindestwert für Retrieval
Pattern-Vorautorisierung Distanz	**<0.3**	Cosine-Distanz zur Vorautorisierungs-Anfrage
Session-Vorautorisierung Default-Dauer	**15 Minuten**	Maximalwert: 60 Minuten, konfigurierbar
Pipeline-Stufen	**4**	Mindestens 3, Referenz-Implementierung 4

Hinweise zum Schutzversprechen (ehrlich): Die Pipeline bietet **keinen 100%-Schutz**. Stufe 1 erkennt definierte strukturierte Muster mit hoher Zuverlässigkeit, soweit diese den implementierten Musterklassen entsprechen; Stufe 2 (LLM-basiert) kann semantische Bezüge zusätzlich abschwächen, ist aber probabilistisch fehlerbehaftet. Die Kombination minimiert Restrisiken auf ein praktikables Maß; eine vollständige Eliminierung ist technisch nicht erreichbar.

Genau aus diesem Grund kommt der menschlichen Freigabeinstanz (HITL, Abschnitt 6.2) eine zentrale Rolle zu: Sie schließt die Lücke, die rein automatische Filter offenlassen, und bindet Verantwortung dort, wo sie hingehört – beim Menschen.

6.2 Human-in-the-Loop (HITL) – Technische Spezifikation

Der Nutzer behält die finale Entscheidungsgewalt über jeden externen Datenfluss aus der iKi heraus. Es geschieht **keine automatische Weiterleitung an externe Dienste ohne explizite Nutzerautorisierung** – entweder als Einzelfreigabe pro Anfrage oder als vorab erteilte und im Audit-Log protokollierte Vorautorisierung.

Trigger-Bedingungen für das HITL-Gate:

Das HITL-Gate wird aktiviert, sobald mindestens eine der folgenden Bedingungen erfüllt ist:

- **(i)** Eine Anfrage zur Übermittlung an cKi vorgesehen ist (Cloud-Versand-Trigger)
- **(ii)** Der lokale PII-Detector einen Confidence-Score über dem konfigurierten Schwellwert zurückgibt
- **(iii)** Der RAG-Retrieval Dokumente mit erhöhter Sensitivitäts-Markierung zurückgibt
- **(iv)** Der Nutzer manuelle Validierung in den Einstellungen aktiviert hat
- **(v)** Der optional aktivierte Ethik-Compliance-Filter (Abschnitt 6.2.1) einen Konflikt mit den Ethik-Grundsätzen der Organisation feststellt

Trigger (i) ist dabei der **architekturell zentrale Trigger**: Jeder Cloud-Versand durchläuft das HITL-Gate, sofern keine vorab erteilte Vorautorisierung greift.

Diff-Anzeige (Nutzer-Übersicht):

Dem Nutzer werden in einer übersichtlichen, menschenlesbaren Darstellung präsentiert:

- Die **Original-Anfrage** (lokal erstellt, mit Klarnamen)
- Die **anonymisierte cKi-Anfrage** (nach Pipeline)
- Die **retrieved RAG-Chunks** (mit RBAC-Filterung) im Original und anonymisiert
- Die **technischen Parameter** der geplanten Übermittlung (Ziel-Modell, Temperatur, Header etc.)
- Visuelles Diff: Entfernte/ersetzte Tokens werden farblich hervorgehoben

Diese visuelle Aufbereitung erfüllt zugleich einen wichtigen Transparenz-Zweck: Dem Nutzer wird unmittelbar deutlich, welche Verarbeitungsschritte im Hintergrund ablaufen und welche Daten konkret nach außen übermittelt würden.

Approval-Modi (alle vom Nutzer einstellbar, jederzeit revidierbar):

- **(a) Single (Default für sensible Kontexte):** Einzelfreigabe pro Anfrage
- **(b) Session (opt-in):** Vorautorisierung für definierte Zeitspanne (Default: 15 Min., max. 60 Min., konfigurierbar)
- **(c) Pattern (opt-in):** Vorautorisierung für strukturell ähnliche Anfragen (Cosine-Embedding-Distanz < 0.3 zur Vorautorisierungs-Anfrage; konfigurierbar). Pattern-Vorautorisierung gilt nur für Anfragen ohne neue PII-Treffer; bei neu erkannter PII fällt das System in Single-Modus zurück.

Nicht-Umgehbarkeit (non-bypassable): Das HITL-Gate kann für jeden Cloud-Versand nicht umgangen werden, sofern keine im Audit-Log protokollierte Vorautorisierung greift. Session- und Pattern-Modi sind explizite, vom Nutzer aktiv eingeschaltete und protokollierte Vorautorisierungen, nicht Umgehungen. Die Aktivierung wie Deaktivierung der Vorautorisierungs-Modi wird im HITL-Audit-Log protokolliert, sodass auch zeitliche Lücken im Einzelfreigabe-Verfahren rückblickend nachvollziehbar bleiben.

Technische Durchsetzung: Die Nicht-Umgehbarkeit ist nicht nur als UI-Konvention zu implementieren, sondern durch **technische Egress-Kontrolle**: Direkte Cloud-API-Calls aus Anwendungs- oder Agentenprozessen werden durch zentrale Gateway-Mechanismen erzwungen; API-Keys liegen ausschließlich beim Orchestrator, nicht in Endnutzer- oder Agentenprozessen.

Das Schalter-Element zum Aktivieren/Deaktivieren der Vorautorisierungs-Modi muss in der Nutzer-Oberfläche jederzeit prominent erreichbar sein; eine farbliche Hervorhebung wird empfohlen.

Bearbeitungsoptionen:

Nach Anzeige der anonymisierten Anfrage kann der Nutzer:

- **(1)** Die Anfrage **direkt bestätigen** und versenden
- **(2)** Die Anfrage in den anpassbaren Feldern **manuell editieren** (Anfrage-Text, RAG-Kontext-Auswahl, Modell-Wahl, Temperatur, weitere Parameter); manuelle Editierungen vor Cloud-Versand werden im Audit-Log mit Hash-Differenz Original ↔ Editiert protokolliert, um nachträgliches Umgehen von PII- oder Ethik-Filterungen nachvollziehbar zu machen
- **(3)** Die Anfrage **ablehnen** (kein Versand an cKi; ggf. Fallback auf reine iKi-Verarbeitung)

6.2.1 Ethik-Compliance-Filter (optional aktivierbar; nicht Teil der obligatorischen Anspruchskette)

Als optional aktivierbare Erweiterung des HITL-Gates kann ein **Ethik-Compliance-Filter** konfiguriert werden, der jede zur Cloud-Übermittlung vorgesehene Anfrage gegen die organisationseigenen Ethik-Grundsätze prüft. Implementierungen ohne aktivierten Filter erfüllen die obligatorische Merkmals-Kombination M1–M5 ebenfalls.

Funktionsweise:

- **Live-Bezug:** Die Ethik-Grundsätze werden zur Laufzeit aus einer definierten Quelle (Organisations-Homepage, Intranet-URL oder lokal gespiegeltes Dokument) geladen, **niemals hartcodiert**. Damit ist sichergestellt, dass stets die aktuell gültige Fassung wirksam ist – ohne Code-Update, ohne System-Neustart.
- **Prüfzeitpunkt:** Die Prüfung erfolgt **vor dem Cloud-Versand**, **nicht** bereits bei reiner iKi-Anfrage. Damit bleibt dem Nutzer der lokale Reflexionsraum erhalten, ohne dass Anfragen ohne externe Konsequenz protokolliert werden.
- **Verstoß-Behandlung:**
  - Anzeige im HITL-Dialog: Anfrage, betroffene Passage und der konkrete Ethik-Grundsatz, gegen den ein Konflikt besteht.
  - Nutzer-Optionen: Anfrage zurückziehen, korrigieren, oder bewusst dennoch freigeben (mit verpflichtender Protokollierung).

- **Manipulationsersichtliches Logging (tamper-evident):** Bei bewusster Freigabe trotz Verstoß wird ein Eintrag in einem Hash-verketteten, append-only Audit-Log geschrieben (siehe Abschnitt 6.6). Dieser Eintrag ist nicht ohne Bruch der Hash-Kette modifizierbar; bei externer Verankerung der Kettenwurzel (z.B. via OpenTimestamps in der Bitcoin-Blockchain) ist jede nachträgliche Manipulation auch durch Administratoren technisch nachweisbar.

Audit- und Compliance-Nutzen:

Eine konsequent aktivierte Ethik-Compliance-Schicht verbindet zwei Aspekte: technische Datensouveränität und institutionelle Selbstverpflichtung. Die Organisation, die diesen Filter aktiviert, koppelt ihre KI-Nutzung **technisch** an die eigenen, öffentlich oder intern dokumentierten Ethik-Grundsätze. Jede Abweichung wird sichtbar, jede bewusste Ausnahme wird protokolliert.

Das hat praktische Auswirkungen:

- **Auditierbarkeit:** Compliance- und QM-Audits können auf eine maschinell auswertbare Entscheidungs-Historie zugreifen, statt nur auf vertragliche Zusicherungen.
- **Konsistenz:** Die Ethik-Grundsätze, an die sich Mitarbeiter halten sollen, gelten technisch auch für die KI-vermittelten Außen-Kommunikationen der Organisation – ohne Sonderregeln.
- **Glaubwürdigkeit:** Eine Organisation, die ihre KI-Cloud-Anfragen technisch gegen ihre eigenen Ethik-Grundsätze prüft und dies manipulationsersichtlich protokolliert, kann diesen Sachverhalt sachlich kommunizieren – etwa gegenüber Geschäftspartnern, Aufsichtsbehörden oder der Öffentlichkeit.

6.2.2 Audit-Log (HITL-Logging)

Jede HITL-Entscheidung wird protokolliert:

- Hash der Original-Anfrage (SHA-256)
- Hash der anonymisierten Anfrage (SHA-256)
- Bei manuellen Editierungen: zusätzlicher Hash der editierten Anfrage und Diff-Marker
- Zeitstempel
- Nutzer-ID
- Entscheidung (Bestätigt / Editiert / Abgelehnt / Vorautorisiert via Session/Pattern / Trotz Ethik-Konflikt freigegeben)
- Modell-Routing (Ziel-cKi)
- Bei Ethik-Konflikt: Referenz auf die geltende Ethik-Grundsatz-Quelle (URL + Hash der Grundsätze zum Zeitpunkt der Anfrage)
- Aktivierung und Deaktivierung von Vorautorisierungs-Modi (Session, Pattern)
- Bei Re-Identifikations-Fehlern (Crash zwischen Cloud-Versand und Re-ID): entsprechender Fehler-Eintrag

Persistierung in append-only Log mit kryptographischer Hash-Verkettung (jeder Eintrag enthält den Hash des Vorgängers). Optionale Erweiterungen zur Erhöhung der Manipulationsresistenz (siehe Abschnitt 6.6):

- Spiegelung auf separates System mit getrennten Admin-Rechten
- Verankerung der Kettenwurzel auf öffentlicher Blockchain via OpenTimestamps oder vergleichbar
- WORM-Speicher (Write-Once-Read-Many)

Aufbewahrung gemäß organisatorischer Vorgaben. Das Log dient als Grundlage für QM-Auditierbarkeit (z.B. ISO 9001) und Datenschutz-Nachweispflichten.

6.2.3 Visualisierung (siehe Abschnitt 11)

Eine grafische Gesamtdarstellung des Datenflusses und der Sicherheitsschichten findet sich in Abschnitt 11 (Architektur-Diagramm).

6.3 Anbieter-Souveränität & Routing-Strategie

Das System vermeidet Abhängigkeiten von einzelnen Anbietern durch aktive Diversifizierung.

- **Modell-Wahl:** Der Implementierer (initial) bzw. der berechtigte Nutzer (laufend) definiert die Regeln für Modell-Hersteller und Anbieter-Regionen (z.B. US-amerikanische, europäische, asiatische Anbieter). Der Orchestrator wendet diese Regeln an.
- **Infrastruktur-Diversifizierung:** Nutzung von Gateways, die Anfragen über mehrere Rechen-Dienstleister nach konfigurierbaren Regeln verteilen. Durch die Verteilung der Anfragen auf verschiedene Rechenstandorte und Anbieter wird die Korrelation einzelner Anfragen zu einem Nutzerprofil statistisch erschwert.
- **ZDR-Auswahlkriterium:** Primäre Auswahl von Anbietern mit Zero-Data-Retention (ZDR)-Richtlinien. Anbieter ohne ZDR können bei expliziter Nutzerentscheidung (z.B. für spezifische Aufgaben mit geringer Sensibilität) konfigurationsspezifisch zugelassen werden, typischerweise über separate API-Schlüssel mit eigenen Guardrail-Profilen (siehe Abschnitt 6.4, Ebene 4a).

6.4 Multi-Level-Guardrails (Richtlinien-Ebenen)

Ebene	Maßnahme	Ziel
1 (ZDR-Policy)	Gateway speichert keine Anfragen zu Trainingszwecken	Inhaltsschutz
2 (Regionale Compliance)	Rechenstandorte innerhalb Jurisdiktionen (z.B. EU-Only)	Datenschutz
3 (Inhaltsfilter)	Zusätzliche PII-Kontrolle auf Gateway-Ebene	Moderation

Ebene	Maßnahme	Ziel
4a (Policy-Kontrolle pro Schlüssel)	Pro API-Schlüssel können individuelle Guardrails (ZDR-Policy, Region, Inhaltsfilter) konfiguriert werden	Differenzierte Sicherheitspolitik
4b (Mandantenfähige Key-Verwaltung)	Soweit von den jeweiligen Anbieter-AGBs gedeckt: organisationsweite Key-Verwaltung mit anbieterseitigem Multi-Workspace-Support; ermöglicht Identitäts-Abstraktion und differenziertes Berechtigungs-Management gegenüber dem Inferenz-Anbieter unter Wahrung der konfigurierten Ebene-1- und Ebene-2-Policies	Identitäts-Abstraktion und Datenschutz
4c (Temporäre Key-Eskalation)	Zeitbasierte Vergabe erweiterter Modell-Berechtigungen (z.B. Zugriff auf Hochleistungs-Modelle) durch Admin-Freigabe für definierte Zeiträume; nach Ablauf automatischer Rückfall auf Standard-Berechtigungen	Kostenkontrolle und Audit-fähige Berechtigungs-Eskalation in Team-Umgebungen

Hinweis zu 4b: Diese Maßnahme setzt die ausdrückliche Zulässigkeit gemäß den Allgemeinen Geschäftsbedingungen des jeweiligen Anbieters voraus. Die Verwendung anbieterseitiger Multi-Workspace-Funktionen ist von einfachem Key-Sharing zu unterscheiden.

6.5 Netzwerk-Sicherheit (begleitende Standardpraxis) & Föderationsschicht (M6)

Folgende netzwerksicherheitsbezogene Maßnahmen werden als begleitende, dem allgemeinen Stand der Technik entsprechende Standardpraxis empfohlen (sie sind nicht als eigenständige Anspruchsmerkmale dieser Architektur zu verstehen):

Maßnahme	Umsetzung
Keine offenen Ports für allgemeinen Internet-Zugriff	Standard-Hardening-Praxis
Verschlüsselte VPN-Verbindungen	z.B. Tailscale, WireGuard oder vergleichbare Lösungen
VLAN-Trennung	Zwischen KI-Systemen, Nutzer-Clients und Server-Infrastruktur
Egress-Kontrolle	Direkter Internetzugriff aus iKi/Agenten-Prozessen wird durch zentrales Gateway erzwungen

Zugriffskontrolle: Authentifizierung und Autorisierung erfolgt zentral über die Plattform mit RBAC-Modellierung (Standardpraxis).

Das **Anspruchsmerkmal M6** bezieht sich auf das **föderierte Inter-Instanz-Protokoll** für sichere Cross-Site-Kommunikation auf Applikationsebene, nicht auf die VPN-Technologie als solche. Während VPN den Transport sichert, sichert die Föderation die Inter-Instanz-Kommunikation auf Applikationsebene – sowohl zwischen NWM-Instanzen mit eigenen iKi-Ressourcen als auch zwischen Instanzen, die iKi-Ressourcen anderer Instanzen über die Föderation mitnutzen.

6.6 Admin-Resistenz & Datenschutz

Problem: Klassische selbstgehostete Plattformen geben dem Administrator technischen Zugriff auf alle Nutzerdaten. Dies kollidiert mit dem Schutzziel der Datensouveränität für Endnutzer in einer Mandanten- oder Mehrnutzerumgebung.

Technische Maßnahmen:

Maßnahme	Beschreibung	Status
Hash-verkettetes append-only Audit-Logging	HITL-Logs (siehe 6.2.2) werden mit kryptographischer Hash-Verkettung gesichert: Jeder neue Eintrag enthält den Hash des Vorgänger-Eintrags. Manipulation eines beliebigen Eintrags invalidiert die Kette ab diesem Punkt und ist damit **manipulationsersichtlich (tamper-evident)**. Optional: Spiegelung auf ein zweites System mit getrennten Admin-Rechten oder Verankerung der Kettenwurzel auf einer öffentlichen Blockchain (OpenTimestamps). Hinweis: Hash-Verkettung macht Manipulation **nachweisbar**, nicht **unmöglich**; technische Unmöglichkeit (tamper-proof) erfordert WORM-Speicher, getrennte Admin-Domänen oder externe Verankerung.	Teil von M3
Client-seitige Verschlüsselung mit nutzereigenen Keys	Sensible Inhalte werden auf Client-Seite mit Nutzer-spezifischen Schlüsseln verschlüsselt, bevor sie auf der SSOT-Plattform abgelegt werden. Der Administrator hat technisch keinen Zugriff auf den Klartext. Empfohlene Werkzeuge: Cryptomator-Vaults, Nextcloud End-to-End-Encryption-App oder vergleichbare Lösungen.	Stark empfohlen
Föderierte Trennung sensibler Bereiche	Über die optionale Föderationsschicht (M6) können Nutzergruppen mit besonders sensiblen Daten eigenständig betriebene Instanzen nutzen, deren Administration getrennt von der Hauptplattform erfolgt. Beispiel: Eine Vorstandsgruppe mit besonders schützenswerten Inhalten betreibt eine eigene NWM-Instanz und föderiert sich nur für	Teil von M6 (optional)

Maßnahme	Beschreibung	Status
	definierte Cross-Site-Funktionen (z.B. Kalender-Abgleich) mit der Hauptplattform; sensible Inhalte bleiben physisch getrennt vom Haupt-Admin.	

Hinweis zur Architektur-Variante mit Client-seitiger Verschlüsselung: Wenn Inhalte clientseitig verschlüsselt auf der SSOT abgelegt werden, ist serverseitiges RAG (Chunking, Embedding, Indexierung) auf diesen Inhalten nicht ohne Weiteres möglich. Diese Defensive Publication offenbart hierfür folgende Ausführungsformen:

- **Variante A:** Serverlesbare Inhalte (kein Client-Encryption) → vollumfängliches Team-RAG
- **Variante B:** Clientseitig verschlüsselte Inhalte mit clientseitiger Indexierung in lokaler Vektor-DB pro Nutzer
- **Variante C:** Nutzerkontrollierte iKi-Instanz mit kontrolliertem Schlüsselzugriff (z.B. via Föderation)
- **Variante D:** Hybrider Ansatz mit serverlesbaren Team-Inhalten und client-verschlüsselten Privatinhalten

Ergänzende organisatorische Maßnahmen:

Maßnahme	Beschreibung
Transparente Kommunikation	Nutzer werden informiert, welche technischen Zugriffsmöglichkeiten der Admin hat und unter welchen Bedingungen Zugriff erfolgt; Logs über Datenzugriffe können auf Wunsch ausgegeben werden.
Vertragliche Absicherung	AGB: Admin verpflichtet sich vertraglich zur Vertraulichkeit; Zugriff ausschließlich im technischen Notfall.
Logging & Audit organisatorischer Zugriffe	Plattform-seitige Protokollierung administrativer Zugriffe, regelmäßige Reviews.

Hinweis: Konkrete Implementierungs-Hinweise und Referenz-Konfigurationen zu diesen Maßnahmen werden im begleitenden GitHub-Repository fortlaufend ergänzt.

6.7 Backup-Konzept (begleitende Standardpraxis)

Das System nutzt die etablierte 3-2-1-Regel als allgemein anerkannte Standardpraxis (kein eigenständiges Anspruchsmerkmal):

Ebene	Maßnahme
3 Kopien	Original + 2 Backups

Ebene	Maßnahme
2 Medien	SSD + HDD (oder Cloud + Lokal)
1 Offsite	Externer Standort

Datenarten:

- Plattform-Daten, persönliche Vaults, Vektor-Datenbanken, HITL-Audit-Logs
- HITL-Audit-Logs erfordern revisionssichere Aufbewahrung gemäß QM-Vorgaben

---

7. Zukunftsszenarien & Erweiterbarkeit

7.1 Mobile Endgeräte & Privacy

Das System ist erweiterbar auf mobile Endgeräte (z.B. stationäre Service-Roboter, autonome Logistikfahrzeuge in geschlossenen Bereichen, mobile Hausassistenten) mit iKi+cKi-Orchestrierung. Privacy-Anforderungen sind dabei identisch: PII-Filter vor Cloud-Sendung, HITL für sensible Aufgaben.

7.2 AGI-Szenario

Bei fundamentalem KI-Fortschritt (AGI) bleibt die Kernarchitektur (lokal + Cloud, HITL, SSOT, Multi-Level-Guardrails) anwendbar. Die Modell-Ebene ist austauschbar; die durch diese Defensive Publication etablierte Prior Art bleibt davon unberührt.

7.3 Skalierung

Größe	Architektur
Einzelnutzer	iKi auf lokalem PC, cKi via Gateway
KMU (<500 MA)	Zentraler Server, mehrere Clients, Team-Berechtigungen
Föderation	Instanzen verbinden sich sicher miteinander – mit oder ohne eigene iKi-Ressourcen pro Instanz → Dezentrales Netzwerk

7.4 Modell-Anpassung und Fine-Tuning

Das System ermöglicht die Integration von feinabgestimmten Modellen (Fine-Tuning) auf Basis lokaler Daten:

- **Lokales Fine-Tuning:** Anpassung von Open-Weight-Modellen (z.B. Llama, Mistral oder Nachfolgegenerationen) auf domänenspezifische Daten unter Beibehaltung der Datensouveränität. Bei aktiviertem Ethik-Compliance-Filter (Abschnitt 6.2.1) werden

auch Fine-Tuning-Datensätze gegen die organisationseigenen Ethik-Grundsätze geprüft, was Drift durch unsauberes Trainingsmaterial vorbeugt.
- **Cloud-basiertes Training:** Nutzung externer Trainingsdienste mit vorheriger Anonymisierung sensibler Daten gemäß Abschnitt 6.1.
- **Modell-Versionierung:** Jedes Fine-Tuning erzeugt eine eindeutig identifizierbare neue Modell-Version mit Metadaten (Trainingsdatum, Trainingsdaten-Hash, Hyperparameter). Bei unerwartetem Verhalten kann auf eine vorherige Version zurückgerollt werden; Verhaltens-Audits werden so reproduzierbar.

Diese Funktionalität erweitert die Architektur um die Möglichkeit, KI-Modelle kontinuierlich an organisationsspezifische Anforderungen anzupassen, ohne die Kernprinzipien der Datensouveränität zu verletzen.

7.5 Dezentrale Netzwerke

Die weiterführende Einbindung dezentraler Energietransfer- und Kommunikationssysteme ist als Erweiterungsoption vorgesehen. Eine entsprechende Anbindung kann insbesondere für Identitäts-, Zeitstempel-, Reputations- und Wertübertragungsfunktionen genutzt werden, ohne die Kernprinzipien der lokalen Datensouveränität zu beeinträchtigen.

7.6 Abgrenzung zum aktuellen Stand der Technik

Im Zeitraum vor und nach Erstveröffentlichung der NWM-Architektur (v3.0, 15.04.2026) wurden mehrere konzeptionell verwandte, jedoch in der Merkmalskombination unterschiedliche Lösungen publiziert:

Quelle / Datum	Überlappung	Abgrenzung zu NWM
Grob github.com/azerozero/grob *22.02.2026*	Regex-basierte PII-DLP, HMAC-Pseudonymisierung, Multi-Provider-Routing auf Gateway-Ebene	Multi-Provider-Routing auf Gateway-Level (Load-Balancing, Failover); keine LLM-basierte semantische Umformulierung; keine Verifikationsstufe; keine RBAC-RAG-Integration auf Applikations-Ebene; HITL nur für Tool-Aufrufe, nicht für Cloud-Versand
OllamØnym github.com/H-Ismael/ollamonym *19.02.2026*	Hybride Erkennung (Regex + lokales LLM), session-stabile Tokens	Zweistufig (Detection + Transformation), keine Verifikation, kein HITL-Gate, keine RBAC-RAG-Integration
Microsoft Presidio *etabliert*	OSS PII-Detection mit Anonymisierung und Re-Identifikation	Reine PII-Tooling-Library; keine hybride iKi/cKi-Orchestrierung; kein HITL-Gate; keine

Quelle / Datum	Überlappung	Abgrenzung zu NWM
		RAG-/RBAC-Integration; keine Multi-Provider-Routing-Schicht
NVIDIA NeMo Guardrails *etabliert*	RAG-Integration, Content Moderation, Tool Gating	Kein lokales LLM als PII-Vorfilter mit semantischer Umformulierung; kein reversibles Session-Mapping; keine Diff-Visualisierung im HITL; nicht für KMU-Selbsthosting konzipiert
Cloak Anonymizer v2.0 anonym.community *14.03.2026*	Native Nextcloud-App für PII-Anonymisierung mit 320+ Entitäten, lokale Verschlüsselung	Statisches File-Processing im Storage; keine RAG-Integration; kein dynamisches Chat-/Agent-Rephrasing; kein Multi-Provider-Routing; kein HITL-Gate
Kong AI PII Sanitizer docs.konghq.com *etabliert*	NER-basierte PII-Erkennung, optional reversibel auf Gateway-Ebene	Gateway-Ebene statt Applikations-Ebene; kein lokales LLM für Umformulierung; Enterprise-Lizenz; kein HITL; keine RBAC-RAG
GuardRAG (Bodea et al., arXiv:2604.15958) *17.04.2026*	Akademisches Evaluations-Framework für Anonymisierungs-Platzierung in RAG	GuardRAG offenbart nicht die konkrete Integration in eine selbstgehostete SSOT mit RBAC-vorgefiltertem Retrieval, HITL-Diff und RAM-only Re-ID-Mapping im hybriden iKi/cKi-Orchestrator. *Hinweis: Nicht zu verwechseln mit GuarantRAG (Zhao et al., arXiv:2604.08046, 09.04.2026), einem thematisch unverwandten Knowledge-Integration-Framework zur Halluzinations-Reduktion in RAG.*
SitePoint Hybrid Cloud-Local LLM Guide sitepoint.com *22.04.2026*	Hybride Routing-Logik (Sensitivity → Complexity → Availability), Multi-Provider-Fallback via LiteLLM	Routing primär auf Modell-Verfügbarkeit und Kosten; nur Regex-PII-Scan; kein reversibles Mapping; kein HITL; keine RBAC-RAG; keine Föderation; keine lokale Sensibilitäts-Klassifikation als Vorprüfung mit nachfolgender mehrstufiger Pipeline
OGX / Securing the Agent (Arceo,	Retrieval-time Authorization, Vendor-neutrales Multi-Tenant-	Enterprise-/Multi-Tenant-orientiert; kein lokales LLM als PII-Vorfilter mit semantischer

Quelle / Datum	Überlappung	Abgrenzung zu NWM
Narsing, arXiv:2605.05287) *06.05.2026*	Routing, Policy-aware Ingestion	Umformulierung; kein reversibles Session-Mapping in flüchtigem Speicher; kein HITL-Diff; kein selbstgehosteter SSOT-Plattformfokus für KMU/Vereine
MemPrivacy (Chen et al., arXiv:2605.09530) *10.05.2026 (v1)*	Edge-basierte PII-Erkennung mit semantisch typisierten Platzhaltern für Cloud-seitige Memory-Verarbeitung; lokale Re-Identifikation	NWM v3.0 (15.04.2026) offenbart bereits typisierte Platzhalter, reversibles RAM-only-Mapping und mehrstufige PII-Pipeline als Bestandteil einer **Gesamt-Architektur** mit HITL-Gate, RBAC-RAG, Multi-Provider-Routing und SSOT-Plattform. MemPrivacy fokussiert auf **Personalized Memory Management** als Komponente; HITL-Gate, RBAC-RAG-Integration, Multi-Provider-Routing und selbstgehostete SSOT-Plattform fehlen. Veröffentlichung 25 Tage nach NWM v3.0.
ArcaQ arcaq.com *etabliert*	100% On-Premise-Architektur, Knowledge-Graph-zentriert, Expert-in-the-Loop, soweit öffentlich ersichtlich umfangreiches Patentportfolio	Kein Hybrid-Ansatz (Air-Gap-First); kein Nextcloud-SSOT; ReBAC für Knowledge Graph statt RBAC-RAG-Retrieval; Enterprise-Produkt

Die NWM-Architektur unterscheidet sich von den vorgenannten Lösungen durch die **spezifische Kombination** aus mehrstufiger PII-Pipeline mit semantischer Umformulierung, RAM-only reversiblem Mapping, HITL-Gate mit Diff-Visualisierung und manipulationsersichtlichem Audit-Logging, optional aktivierbarem Ethik-Compliance-Filter, RBAC-bewusster RAG-Retrieval-Filterung, ZDR-Multi-Provider-Routing, technischer Admin-Resistenz und optionaler Föderation – integriert in eine selbstgehostete SSOT-Plattform für die Zielgruppe KMU/Verein/Einzelanwender. Insbesondere v3.0 vom 15.04.2026 etabliert die hybride Orchestrierungs-Grundkonzeption als Prior Art zeitlich vor dem SitePoint-Guide (22.04.2026), den OGX-Veröffentlichungen (06.05.2026) und MemPrivacy (10.05.2026).

---

8. Lizenzierung & Patentschutz

Aspekt	Spezifikation
Lizenz für Dokumentation	CC BY-NC-SA 4.0
Wirkung	Dritte müssen Urheber nennen, keine kommerzielle Nutzung des Dokuments, gleiche Lizenz für Abwandlungen
Patentschutz (Prior Art)	Defensive Publication mit Zeitstempel-Verankerung über Zenodo (DOI), GitHub (Release-Tag), eigene Domain, Wayback-Machine sowie Bitcoin-Blockchain-Verankerung des PDF-Hashes via OpenTimestamps (siehe Header dieses Dokuments und begleitende `.ots`-Datei im Repository)
GitHub-Repository	github.com/Netz-Werkzeug-Macherei/nwm-defpub – begleitet die Defensive Publication mit Implementierungs-Hinweisen und Referenz-Konfigurationen
Homepage	netz-werkzeug-macherei.de – primärer Veröffentlichungsort des Projekts
Zenodo-Veröffentlichung	Auf Zenodo wird ausschließlich das Defensive-Publication-Dokument selbst (nebst zugehöriger `.ots`-Datei) hinterlegt; nicht die begleitende Implementierungs-Dokumentation
Vorgängerversion v3.0	Veröffentlicht 15.04.2026 (DOI 10.5281/zenodo.19597550) – sichert die Kernarchitektur als Prior Art
Vorgängerversion v3.2	Veröffentlicht 22.04.2026 (DOI 10.5281/zenodo.19699717) – erweitert die technische Beschreibung

Wichtig: Nicht die Lizenz schützt vor Patenten, sondern die **Veröffentlichung selbst** (Zeitstempel). Die CC BY-NC-SA 4.0 Lizenz gewährt keine Patentlizenzen.

Hinweis zum begleitenden GitHub-Repository: Das Repository wird als lebendes Begleitwerk fortlaufend mit Implementierungs-Beispielen, Setup-Skripten, Architektur-Diagrammen und FAQ ergänzt. Stand und Inhalte können vom Stand dieser Defensive Publication abweichen. Die hier offenbarten technischen Lehren der Defensive Publication sind eigenständig enabling und ergeben sich vollständig aus dem Text dieser Schrift.

Haftungsausschluss und rechtliche Hinweise:

1. Diese Veröffentlichung beschreibt ein Architekturkonzept und stellt keine fertige Software-Lösung dar. Die tatsächliche Sicherheit im Betrieb hängt von der korrekten Implementierung, der Einhaltung von Sicherheitsrichtlinien durch den Betreiber sowie der fortlaufenden Anpassung an aktuelle Bedrohungslagen und regulatorische Anforderungen ab.
2. Alle genannten Produkte, Dienstleistungen und Technologien dienen als Referenz-Implementierungen und sind Warenzeichen oder eingetragene Warenzeichen ihrer jeweiligen Eigentümer. Die Nennung stellt keine Produktempfehlung dar.

3. Diese Veröffentlichung dient ausschließlich der Sicherung des Standes der Technik (Prior Art) und stellt keine Patentlizenz dar. Die CC BY-NC-SA 4.0 Lizenz bezieht sich auf das Urheberrecht an dieser Dokumentation, nicht auf eventuelle Patentrechte, die Dritte an den beschriebenen Verfahren oder Systemen halten könnten.
4. Keine Gewährleistung für Vollständigkeit, Richtigkeit oder Aktualität der genannten Preise, Modellversionen, regulatorischen Verweise oder der im begleitenden GitHub-Repository veröffentlichten Implementierungs-Hinweise.
5. Die rechtliche Zulässigkeit konkreter Implementierungen – insbesondere bezüglich DSGVO, EU AI Act und der Allgemeinen Geschäftsbedingungen externer KI-Dienstleister – ist im Einzelfall durch den Implementierer zu prüfen.

---

9. Schlussfolgerung

Durch diese Veröffentlichung wird der Stand der Technik/Wissenschaft für eine Wissensmanagement-Systemarchitektur mit hybrider KI-Integration für datensouveräne Organisationen und Einzelanwender mit dezentraler Netzwerk-Funktionalität etabliert.

Das spezifische, sequenzielle Zusammenspiel der Merkmale M1–M7 (siehe Abschnitt 3) – insbesondere:

- **Lokaler iKi-Orchestrierung mit mehrstufiger PII-Filter-Pipeline (M1, M2)**
- **Konfigurierbarem HITL-Gate mit Trigger-Mechanismus, Diff-Visualisierung, Audit-Logging und optional aktivierbarem Ethik-Compliance-Filter (M3)**
- **RAG-Anbindung an selbstgehostete SSOT-Plattform mit RBAC-Berücksichtigung im Retrieval-Prozess in Vor- oder Nachfilterungs-Variante (M4)**
- **Multi-Provider-Routing mit ZDR-Policy, mandantenfähiger Key-Verwaltung und temporärer Key-Eskalation (M5)**
- **Optionaler Föderationsschicht für sichere Inter-Instanz-Kommunikation – mit oder ohne eigene iKi-Ressourcen pro Instanz (M6)**
- **Optionaler, modular zuschaltbarer Multimodal-Erweiterung (M7a–M7d) sowie Fine-Tuning-Integration (Abschnitt 7.4) unter Beibehaltung der Sicherheitskette**
- **Technischer Admin-Resistenz durch manipulationsersichtliches Hash-verkettetes Audit-Logging (M3) und ergänzende Client-seitige Verschlüsselung sowie optionale Blockchain-Verankerung der Log-Kettenwurzel**

stellt eine Lösung für das Problem der Datensouveränität bei gleichzeitiger Nutzung moderner KI-Leistung dar.

Als auditierbare Orchestrierungsarchitektur unterstützt das System Qualitätsmanagement-Anforderungen (Auditierbarkeit, Konsistenz, Dokumentation) und ermöglicht die technische Abbildung von Prozessstandards (z.B. ISO 9001) auf technischer Ebene.

---

10. Quellen & Referenzen

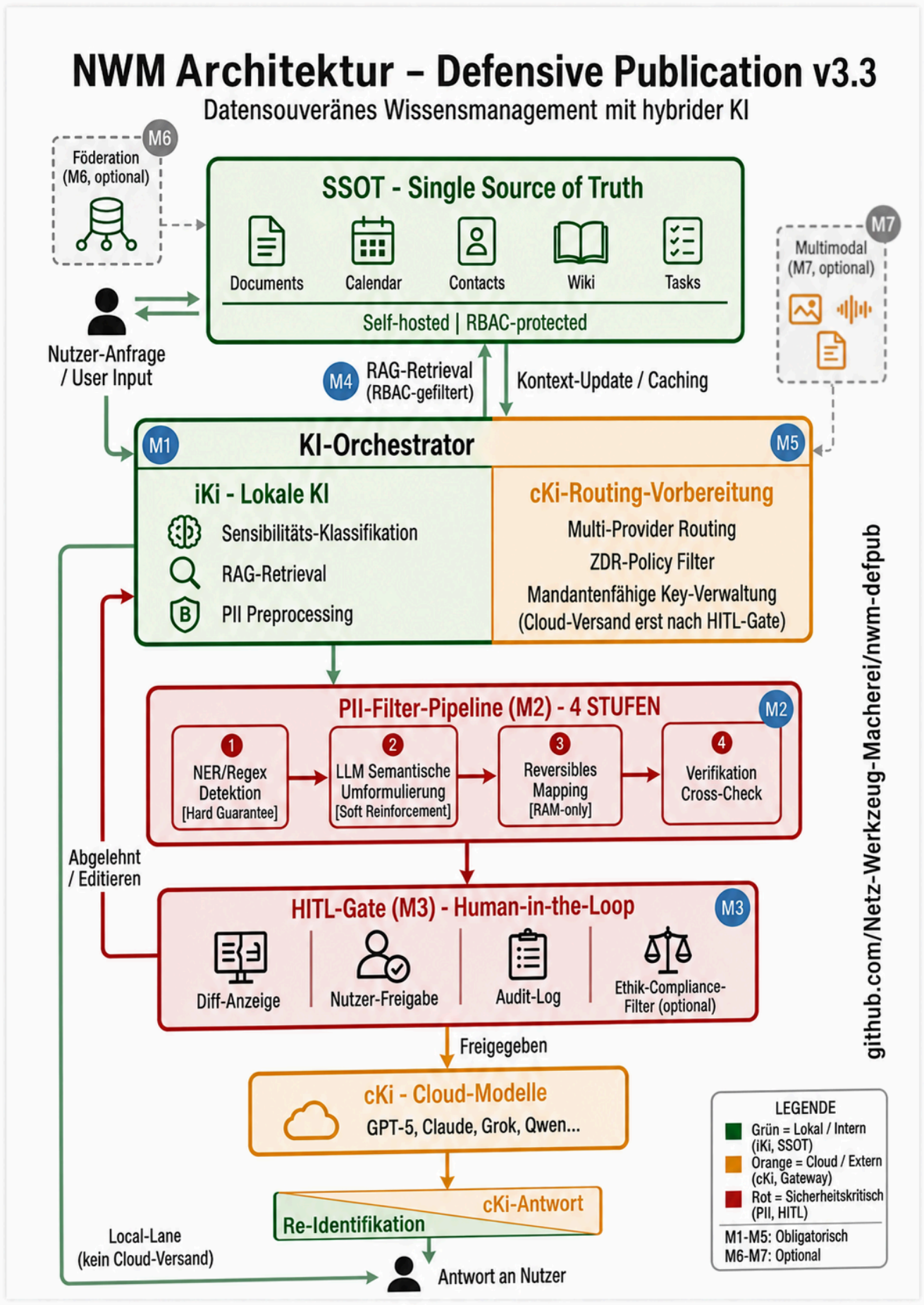
ID	Quelle	Beschreibung	Link
[1]	NWM v3.0 (Vorgänger)	Defensive Publication v3.0, 15.04.2026	DOI 10.5281/zenodo.19597550 · Wayback
[2]	NWM v3.2 (Vorgänger)	Defensive Publication v3.2, 22.04.2026	DOI 10.5281/zenodo.19699717 · Wayback HP 22.04.2026 · Wayback PDF 24.04.2026
[3]	NWM GitHub-Repository	Begleitende Implementierungs-Dokumentation	github.com/Netz-Werkzeug-Macherei/nwm-defpub
[4]	Grob	Rust-basierter LLM-Proxy mit DLP/PII-Pseudonymisierung	github.com/azerozero/grob
[5]	OllamØnym	Hybride PII-Detection-Engine (Regex + lokales LLM)	github.com/H-Ismael/ollamonym
[6]	Microsoft Presidio	Open-Source PII-Detection und Anonymisierung	github.com/microsoft/presidio
[7]	NVIDIA NeMo Guardrails	OSS Framework für RAG-Guardrails und Tool-Gating	github.com/NVIDIA/NeMo-Guardrails
[8]	Cloak Anonymizer v2.0	Nextcloud-native PII-Anonymisierung, 14.03.2026	anonym.community
[9]	Kong AI PII Sanitizer	Enterprise-Gateway-Plugin für PII-Anonymisierung	docs.konghq.com/hub/kong-inc/ai-sanitizer/
[10]	GuardRAG (Bodea et al., IWSPA '26)	Anonymisierungs-Evaluations-Framework für RAG-Pipelines	arXiv:2604.15958
[11]	SitePoint Hybrid Cloud-Local LLM Guide	Architektur-Guide für hybride KI-Infrastruktur, 22.04.2026	sitepoint.com/hybrid-cloudlocal-llm-the-complete-architecture-guide-2026/
[12]	OGX / Securing the Agent (Arceo, Narsing)	Vendor-neutrales Multi-Tenant Enterprise Retrieval, 06.05.2026	arXiv:2605.05287
[13]	MemPrivacy (Chen et al.)	Privacy-Preserving Personalized Memory für Edge-Cloud Agents, 10.05.2026	arXiv:2605.09530

ID	Quelle	Beschreibung	Link
[14]	ArcaQ	Sovereign Decision Intelligence Platform	arcaq.com
[15]	OpenRouter	Multi-Model-Gateway (Referenz-Implementierung)	openrouter.ai
[16]	DSK Orientierungshilfe RAG	Datenschutzrechtliche Besonderheiten generativer KI-Systeme (Okt 2025)	datenschutzkonferenz-online.de
[17]	EU AI Act / KI-VO	Art. 4 KI-VO; EU AI Omnibus (ab 02.08.2026)	digital-strategy.ec.europa.eu
[18]	Ollama	Lokale Modellausführung	ollama.com
[19]	Nextcloud Assistant / LLM2 / Context Chat	Offizielle Nextcloud-KI-Apps	nextcloud.com/assistant/
[20]	LangChain	Workflow-Framework für KI-Agenten	langchain.com
[21]	AutoGen	Multi-Agenten-Systeme	microsoft.github.io/autogen
[22]	CrewAI	Rollen-basierte Agenten-Teams	crewai.com
[23]	ChromaDB	Vektor-Datenbank für RAG	trychroma.com
[24]	Qdrant	Vektor-Datenbank für RAG	qdrant.tech
[25]	Tailscale	VPN für sichere Verbindungen	tailscale.com
[26]	WireGuard	VPN-Protokoll	wireguard.com
[27]	Cryptomator	Client-Side-Verschlüsselung	cryptomator.org
[28]	spaCy	NER-Bibliothek für PII-Erkennung	spacy.io
[29]	OpenTimestamps	Bitcoin-Blockchain-basierte Zeitstempel-Verankerung	opentimestamps.org

---

11. Architektur-Diagramm

Die nachfolgende grafische Darstellung visualisiert den Datenfluss und die Sicherheitsschichten der NWM-Architektur. Sie ergänzt die textliche Spezifikation der vorhergehenden Abschnitte und dient als kompakte Übersicht für Implementierer und Entscheider.



Bildunterschrift:

Abbildung 1: Sequenzieller Datenfluss der NWM-Architektur. Eingehende Nutzeranfragen durchlaufen die lokale iKi-Schicht (Sensibilitäts-Klassifikation, RAG-Retrieval, mehrstufige PII-Pipeline), gefolgt vom HITL-Gate mit optionalem Ethik-Compliance-Filter, vor einer eventuellen Cloud-Übermittlung an die cKi-Schicht. Cloud-Antworten werden lokal re-identifiziert und an den Nutzer ausgegeben. Optionale Erweiterungen (Föderation M6, Multimodal M7) sind seitlich angedeutet. Kernmerkmale der bevorzugten Ausführungsform (M1–M5) durchgezogen, optionale Erweiterungen (M6, M7) gestrichelt dargestellt.

Verweis-Mapping: Die im Diagramm dargestellten Bezeichnungen (M1–M7, Stufen 1–4 der PII-Pipeline, Trigger (i)–(v) des HITL-Gates) korrespondieren eindeutig mit den entsprechenden Abschnitten dieses Dokuments und ermöglichen so den schnellen Wechsel zwischen visueller Übersicht und textlicher Detailspezifikation.

Hinweis zur Verwendung des Diagramms:

- Das Diagramm steht unter derselben Lizenz (CC BY-NC-SA 4.0) wie das vorliegende Dokument.
- Hochauflösende Versionen (PDF, PNG) werden im begleitenden GitHub-Repository bereitgestellt: github.com/Netz-Werkzeug-Macherei/nwm-defpub.
- Das Diagramm darf eigenständig (z.B. als Poster im A3-Format) verbreitet werden, sofern Lizenz und Urhebernennung gewahrt bleiben.

Diese Veröffentlichung erweitert v3.2 (22.04.2026) und konsolidiert Implementierungs-Spezifikationen. Vorgängerversionen v3.0 und v3.2 bleiben als Zeitstempel-Anker für Prior-Art-Zwecke wirksam.
