Technische Hochschule
Brandenburg
University of
Applied Sciences

Konzeption und Architekturmodell zur MCP-Integration in CMS anhand von TYPO3 für KI-kompatible Informationsbereitstellung im öffentlichen Sektor

Bachelorarbeit

zur Erlangung des Grades Bachelor of Science
des Fachbereichs Informatik und Medien der
Technischen Hochschule Brandenburg

vorgelegt von:
Philip Seibold

Betreuer: Sebastian Kreideweiß, M.Sc.
Zweitgutachter: Prof. Dr. rer. nat. Martin Christof Kindsmüller

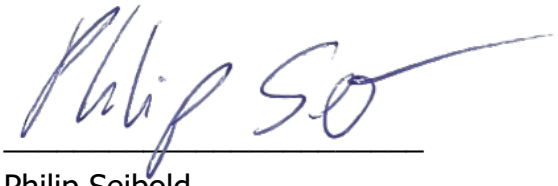
Jena, 26. September 2025

Inhalt

Eidesstattliche Erklärung

Hiermit erkläre ich, dass ich, Philip Seibold, die vorliegende Arbeit selbstständig verfasst habe, dass ich sie zuvor an keiner anderen Hochschule als Prüfungsleistung eingereicht habe und dass ich keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Alle Stellen der Arbeit, die wörtlich oder sinngemäß aus Veröffentlichungen oder aus anderweitigen fremden Äußerungen entnommen wurden, sind als solche kenntlich gemacht.

Jena, 26.09.2025



Philip Seibold

Abkürzungsverzeichnis

API	Application Programming Interface
BFSG	Barrierefreiheitsstärkungsgesetz
BITV	Barrierefreie Informationstechnik Verordnung
BSI	Bundesamt für Sicherheit in der Informationstechnik
CMS	Content Management System
CRUD	Create Read Update Delete
DSGVO	Datenschutz-Grundverordnung
DSK	Datenschutzkonferenz
ENISA	European Union Agency for Cybersecurity
JSON	JavaScript Object Notation
JSON-LD	JSON for Linked Data
JSON-RPC	Remote Procedure Call mit JSON
JWT	JSON Web Token
LLM	Large Language Model
MCP	Model Context Protocol
NLP	Natural Language Processing
OAuth 2.0	Framework für Autorisierung
OIDC	OpenID Connect
PSR-3	PHP Standards Recommendation 3, Logger Interface
RAG	Retrieval Augmented Generation
RDF	Resource Description Framework
RFC	Request for Comments
STDIO	Standard Input Output
W3C	World Wide Web Consortium
WCAG	Web Content Accessibility Guidelines
WebSockets	WebSocket-Protokoll
WHATWG	Web Hypertext Application Technology Working Group

Abbildungsverzeichnis

1. Einleitung

1.1 Kontext und Motivation

In einer zunehmend digitalisierten Verwaltung und öffentlichen Informationsbereitstellung wächst der Bedarf an intelligenten Systemen, die nicht nur Daten speichern, sondern auch kontextualisiert zugänglich machen. Klassische Content-Management-Systeme (CMS) wie TYPO3 stoßen dabei an Grenzen, da Inhalte primär zur menschlichen Rezeption strukturiert werden. Gleichzeitig gewinnen KI-gestützte Verfahren wie Retrieval-Augmented Generation (RAG) an Bedeutung, die externe Wissensquellen zur Verbesserung von Antworten dynamisch einbeziehen. Damit solche Systeme zuverlässig funktionieren, benötigen sie Zugriff auf strukturierte, semantisch angereicherte Inhalte. Es ist derzeit davon auszugehen, dass dieser Trend sich weiter verstärken wird.

Laut einer repräsentativen Studie von Bitkom[1] (2025) haben bereits 67 % der deutschen Bevölkerung generative KI-Anwendungen wie ChatGPT oder Gemini genutzt. Die Tendenz ist dabei steigend. KI-Systeme ersetzen zunehmend klassische Recherchewege über Suchmaschinen und Webportale. Allerdings zeigen Medizinische Studien Halluzinationsraten von bis zu 39,6 % bei GPT-3.5**,** 28,6 % bei GPT-4 und in spezifischen Fällen bis zu 91,4 % bei Gemini (ehemals Bard)[2].

Sie reichen von ungenauen Daten bis hin zu komplett falschen Antworten. Für den öffentlichen Sektor ist das ein Risiko. Bürger*innen verlassen sich auf korrekte, aktuelle und nachvollziehbare Inhalte. Werden diese über KI-Systeme fehlerhaft transportiert, drohen Vertrauensverlust, Fehlentscheidungen oder gar Desinformation. Es ergibt sich somit eine dringende Notwendigkeit, öffentlich bereitgestellte Inhalte systematisch KI-kompatibel aufzubereiten.

---

[1] Bitkom e. V., „KI-Nutzung boomt – aber die Angst vor Abhängigkeit vom Ausland ist groß“, Bitkom, 5. Mai 2025, https://www.bitkom.org/Presse/Presseinformation/KI-Nutzung-boomt-Angst-vor-Abhaengigkeit-Ausland-gross zuletzt geprüft am 11.09.2025, 09:13 Uhr.

[2] Mikaël Chelli u. a., „Hallucination Rates and Reference Accuracy of ChatGPT and Bard for Systematic Reviews: Comparative Analysis“, *Journal of Medical Internet Research* 26 (Mai 2024): e53164, https://doi.org/10.2196/53164.

Während das Model Context Protocol (MCP) im Drupal-Ökosystem bereits erste Wege zur strukturierten Kontextbereitstellung aufzeigt, fehlt es aktuell für TYPO3 an vergleichbaren Ansätzen. Die Integration semantischer Informationsmodelle und standardisierter Schnittstellen zur KI-Nutzung steht hier noch am Anfang. Daraus ergibt sich eine Forschungslücke, die diese Arbeit adressieren möchte.

Das CMS TYPO3 wurde aufgrund der aktuell begonnen Migration[3] von 90+ Bedarfsträgern auf Bundesebene in Deutschland mit über 500 Websites hin zur bundeseigenen CMS-Lösung GovernmentSiteBuilder (kurz GSB) Version 11 auf Basis TYPO3 (Version 12) ausgewählt. Eine Optimierung des Zusammenspiels der Inhalte aus dem CMS in anfragende KI-Lösungen oder umgekehrt, kann eine signifikante Optimierung im Bereich der Verwaltungsdigitalisierung bedeuten. Ziel dieser Arbeit ist es, ein konzeptionelles Modell zu entwickeln, wie MCP in TYPO3 integriert werden kann, um Inhalte KI-gerecht bereitzustellen. Darüber hinaus eröffnet das konzeptionelle Modell Perspektiven für die zukünftige Entwicklung von TYPO3-Extensions, die auf Basis von MCP und KI deutlich spezifischere und kontextabhängige Funktionalitäten bereitstellen können.
Daraus ergibt sich für diese Arbeit die Notwendigkeit, spezifische Forschungsfragen aufzustellen, die im nächsten Abschnitt (Kapitel 1.2) formuliert werden.

1.2 Forschungsfragen

Ausgehend von der dargestellten Problemstellung und den damit verbundenen Anforderungen an Content-Management-Systeme im Kontext Künstlicher Intelligenz ergeben sich folgende zentrale Forschungsfragen, die im Rahmen dieser Arbeit untersucht werden sollen:

F1: Wie kann das Model Context Protocol (MCP) konzeptionell genutzt werden, um den öffentlichen Sektor bei nachvollziehbarer und vertrauenswürdiger KI-Kommunikation zu unterstützen?

[3] Materna Information & Communications SE, „Digitale Souveränität stärken: Materna und leistungsstarkes Partnernetzwerk gewinnen Rahmenvertrag“, Materna, 22. Mai 2025, https://www.materna.de/newshub/presse/digitale-souveraenitaet-staerken-materna-und-leistungsstarkes-partnernetzwerk-gewinnen-rahmenvertrag/ zuletzt geprüft am 11.09.2025, 09:13 Uhr.

Diese Frage zielt darauf ab, das Potenzial des MCP für öffentliche Institutionen genauer zu erfassen und zu ermitteln, inwiefern MCP-Lösungen relevante Anforderungen wie Datenschutz, Barrierefreiheit und Nachvollziehbarkeit erfüllen können.

> F2: Wie kann die Integration des Model Context Protocols (MCP) in TYPO3 gestaltet werden, um die strukturelle Anschlussfähigkeit an Retrieval-Augmented-Generation-Systeme (RAG) zu erhöhen?

Ziel dieser Forschungsfrage ist es, eine theoretische Systemarchitektur zu erarbeiten, die MCP und TYPO3 optimal miteinander verbindet. Dabei sollen konzeptionelle Szenarien skizziert werden, die eine praxisorientierte Implementierung ermöglichen und zugleich die Einbindung in moderne KI-Systeme unterstützen.

> F3: Welche fachlichen, technischen und governancebezogenen Voraussetzungen sind für eine Integration des Model Context Protocols in TYPO3 erforderlich, damit KI-Funktionen kontrolliert an Redaktionsprozesse gekoppelt werden können?

Mit dieser Forschungsfrage wird in qualitativen Expertenbefragungen erhoben, welche Bedingungen in TYPO3 als notwendig gelten. Im Fokus stehen Workflows mit Entwurfsarbeit und nachgelagerter Freigabe, ein fein abgestuftes Rollen- und Rechtemodell, konsequente Versionierung, lückenlose Protokollierung, klar definierte Schnittstellenstandards und zentral gepflegte Tool Verträge. Zusätzlich werden Anforderungen aus dem öffentlichen Umfeld berücksichtigt. Darunter Datenschutz, Transparenz, Barrierefreiheit und Nachvollziehbarkeit. Ziel ist die Einschätzung der praktischen Umsetzbarkeit dieser Voraussetzungen im realen Redaktionsbetrieb.

Die Beantwortung dieser Forschungsfrage bildet die Grundlage für die Entwicklung und Bewertung des Architekturkonzeptes zur MCP-Integration in TYPO3, dass im weiteren Verlauf dieser Arbeit vorgestellt wird.

1.3 Zielsetzung und Methodik

Das primäre Ziel dieser Bachelorarbeit ist es, ein konzeptionelles Architekturmodell zu entwickeln, das zeigt, wie das Model Context Protocol (MCP) effektiv in das Content-Management-System TYPO3 integriert werden kann, um eine KI-kompatible und semantisch angereicherte Inhaltsbereitstellung zu gewährleisten. Dabei wird der öffentliche Sektor als relevantes Anwendungsszenario betrachtet, da dieser spezifische Anforderungen hinsichtlich Transparenz, Autorität, Datenschutz, Barrierefreiheit und Nachvollziehbarkeit an KI-gestützte Informationsangebote stellt.

Zur Erreichung dieses Ziels werden zunächst im Rahmen eines technischen und kontextuellen Literaturüberblicks die theoretischen Grundlagen geschaffen. Dieser Überblick umfasst die Einordnung von TYPO3 und CMS-Systemen in den aktuellen KI-Kontext, eine Analyse relevanter semantischer Datenformate sowie eine Einführung in die Funktion und Rolle von Retrieval Augmented Generation (RAG) und MCP. Anschließend erfolgt eine qualitative empirische Untersuchung in Form von Expertenbefragungen. Hierbei wurden zehn Fachpersonen aus dem erweiterten TYPO3-, CMS- und KI-Umfeld mittels eines standardisierten Fragebogens befragt. Ziel der Befragungen ist es, Einschätzungen zur Umsetzbarkeit und Akzeptanz der MCP-Integration zu sammeln. Gleichzeitig werden die dafür nötigen organisatorischen und technischen Voraussetzungen in TYPO3 erhoben, darunter Workflows mit Entwürfen und Freigaben, ein differenziertes Rollen- und Rechte-Modell, Versionierung, vollständige Protokollierung, klare Schnittstellenstandards und zentral gepflegte Tool-Verträge. Ebenso spielen Datenschutz, Transparenz, Barrierefreiheit und Nachvollziehbarkeit eine Rolle. Die Ergebnisse dieser Befragungen fließen direkt in die Konzeptentwicklung ein und dienen dazu, die Praxistauglichkeit und Realisierbarkeit der theoretischen Architektur zu überprüfen.

Auf Basis der gewonnenen theoretischen Erkenntnisse und empirischen Ergebnisse erfolgt anschließend die Entwicklung eines konkreten Konzeptmodells. Dieses Modell umfasst die Beschreibung notwendiger technischer Komponenten, Schnittstellen und konzeptioneller Implementierungsszenarien für MCP innerhalb von TYPO3. Abschließend wird das entwickelte Architekturkonzept einer kritischen Analyse und Bewertung unterzogen, in der insbesondere Chancen, Herausforderungen und die

Abgrenzung zu bestehenden Lösungen beleuchtet werden. Die Arbeit konzentriert sich auf zwei Wege, nämlich das Lesen und das Entwerfen. Im Leseweg stellt ein MCP-Client autorisierte TYPO3 Inhalte in reduzierter Form bereit. Für Aufgaben mit hohem Wissensbedarf kann zusätzlich die Headless Ausgabe als Korpus für Retrieval Augmented Generation genutzt werden. Im Entwurfsweg erstellt die Assistenz nur Entwürfe in Workspaces. Menschen prüfen diese Entwürfe und geben sie über den bekannten Freigabeprozess frei. Versionierung und Logging halten jeden Schritt fest. Veröffentlichen bleibt eine redaktionelle Entscheidung. MCP liefert dafür die passenden Schnittstellen und Verträge. So rücken Governance, Datenminimierung und Auditierbarkeit in den Mittelpunkt und bereiten die in Kapitel 5 beschriebenen Referenzflüsse sowie das Architekturmodell vor.

1.4 Aufbau der Arbeit

Die Gliederung dieser Arbeit orientiert sich an einem klaren Aufbau: Kapitel 1 führt in Problemstellung, Zielsetzung und Forschungsfragen ein, verortet das Thema im Kontext öffentlicher Einrichtungen und grenzt den Untersuchungsrahmen ab. Kapitel 2 behandelt die theoretischen Grundlagen und analysiert alle relevanten Komponenten im Hinblick auf ihre Funktionsweise und ihre Einordnung in das übergeordnete Architekturmodell.
Kapitel 3 widmet sich der Erhebung von Praxisinteresse an einer MCP-Implementierung. Hierzu werden zehn Expert*innen mittels eines standardisierten Fragebogens befragt. Die Ergebnisse fließen in die Bewertung der Relevanz und Machbarkeit ein. Kapitel 4 berichtet die Ergebnisse der Befragung. Kapitel 5 entwickelt darauf aufbauend das Konzept zur MCP-Integration in TYPO3 und eine exemplarische Systemarchitektur. Kapitel 5 überprüft auch, inwiefern die zuvor definierten Anforderungen durch das Konzept erfüllt werden konnten, und stellt vergleichbare bestehende Ansätze gegenüber. Kapitel 6 bildet den Abschluss der Arbeit. Da es sich um einen theoretischen Ansatz handelt, wird hier ein Ausblick auf mögliche Weiterentwicklungen gegeben und das Potenzial einer praktischen Umsetzung diskutiert.

2. Kontextueller und technischer Literaturüberblick

Kapitel 2 bietet einen strukturierten Überblick über zentrale technologische und konzeptionelle Grundlagen, die für die spätere Konzeptentwicklung relevant sind. Die Darstellung orientiert sich an bestehenden Forschungs- und Entwicklungsarbeiten aus dem CMS- und KI-Kontext. Im Vordergrund stehen dabei konkrete technische Standards, Integrationsszenarien für KI in Content-Management-Systeme sowie bestehende Architekturen wie das Model Context Protocol. Ziel ist es, den aktuellen Stand praxisrelevanter Technologien zusammenzufassen und bestehende Lücken in der Umsetzung, insbesondere im TYPO3-Umfeld, sichtbar zu machen.

2.1 TYPO3 und CMS im KI-Kontext

„Artificial Intelligence is no longer just a trend — it's becoming a key component in many CMS environments."[4]

Diese Einschätzung aus dem TYPO3-Umfeld verdeutlicht den zunehmenden Einfluss KI-basierter Technologien auf moderne Content-Management-Systeme. Während CMS ursprünglich für die menschliche Nutzung optimiert wurden, stellt die Integration von Künstlicher Intelligenz neue Anforderungen an Struktur, Schnittstellen und semantische Tiefe von Inhalten. Anforderungen betreffen die Struktur, die Schnittstellen und die semantische Tiefe von Inhalten. TYPO3 als etabliertes Open-Source-CMS mit starker Verbreitung im öffentlichen Sektor dient dabei als Beispiel, wie ein bestehendes System so weiterentwickelt werden kann, dass es Inhalte nicht nur verwaltet, sondern auch maschinenlesbar und kontextualisiert bereitstellt.

In diesem Kapitel wird untersucht, wie sich TYPO3 historisch und aktuell zwischen der klassischen Inhaltsverwaltung und den Anforderungen generativer KI positioniert und welche Entwicklungen sich daraus ergeben.

Bereits im Jahr 2019 lassen sich erste dokumentierte Berührungspunkte von TYPO3 mit KI-Technologien nachweisen. In einem Beitrag im TYPO3-Blog (Februar 2019) wurde die geplante Anbindung des NLG-Dienstes textengine.io

[4] Frank Nägler, „TYPO3 V14: Building a System for Community-Driven AI Integrations“, TYPO3, 29. Juni 2025, https://typo3.org/article/typo3-v14-ai-integrations.

(Retresco) an TYPO3 vorgestellt[5]. Ziel dieser Integration war es, automatisiert Inhalte wie Sport- oder Wetterberichte zu generieren. Ein Ansatz, der gemeinhin als „Roboterjournalismus" bezeichnet wird. Die TYPO3 GmbH sprach in diesem Zusammenhang explizit von „Artificial Intelligence (AI)", wodurch deutlich wurde, dass das Thema KI bereits früh als relevant für das Ökosystem erkannt wurde. Die Umsetzung basierte auf einer Schnittstelle zu Retresco, einem deutschen Anbieter für KI-basierte Sprachgenerierung. Obwohl es sich bei diesen frühen Ansätzen noch nicht um tiefgreifende semantische Architekturen handelte, markieren sie den Einstieg in eine fortschreitende Auseinandersetzung mit KI-Technologien im TYPO3-Umfeld. In den darauffolgenden Jahren wurde diese Entwicklung nicht direkt in den Kern des Systems überführt, sondern zunehmend in Form externer Erweiterungen und individueller Schnittstellenlösungen weitergedacht.

Im Jahr 2025 ist das Thema KI ein global relevantes Thema und eines der aktuell wichtigsten Forschungsgebiete. Auch TYPO3 befindet sich heute in einer anderen Ausgangslage als noch 2019 und hat seine aktuellen Ziele im Umgang mit künstlicher Intelligenz im eingangs zitierten Beitrag von Frank Nägler veröffentlicht. Im Kern des CMS ist keine direkte KI vorgesehen. Die aktuelle Philosophie und Herangehensweise an KI wird sich in Form von Interfaces also Benutzeroberflächen widerspiegeln. Aufgrund der vielseitigen Anforderungen von TYPO3-Nutzer*innen verfolgt das System keinen monolithischen Ansatz, sondern setzt auf offen definierte Schnittstellen, über die externe KI-Dienste oder eigene Modelle eingebunden werden können. Dieser modulare Weg ermöglicht eine flexible Integration unterschiedlicher KI-Services, unabhängig davon, ob es sich um cloudbasierte Lösungen etablierter Anbieter oder um lokal gehostete, datenschutzkonforme Modelle handelt.

Ziel ist es, die schnelle technologische Entwicklung im KI-Bereich abzubilden, ohne sich auf einzelne Technologien festzulegen oder Nutzer*innen in proprietäre Abhängigkeiten zu bringen. Vielmehr sollen Inhalte kontextbewusst, zugriffsberechtigt und redaktionssicher verarbeitet werden können. Das betrifft insbesondere die Herausforderungen im Umgang mit Arbeitsbereichen (Workspaces), Mehrsprachigkeit, komplexen Seitenstrukturen und differenzierten Benutzerrechten.

[5] Sebastian Küchenmeister, „Natural Language Generation: Using AI in Content Creation - TYPO3 the Open Source Enterprise CMS", 28. Februar 2019, https://typo3.com/blog/natural-language-generation-using-ai-in-content-generation letzter Zugriff 12.09.2025 12:10 Uhr.

Dabei rückt die Notwendigkeit in den Fokus, präzise zu definieren, welche Informationen einer KI zur Verfügung gestellt werden dürfen, um sinnvolle Ergebnisse zu liefern, ohne die Sicherheit oder den Datenschutz zu gefährden. Die aktuell laufenden konzeptionellen Arbeiten zielen deshalb auf ein robustes Rahmenwerk ab, das diese Anforderungen systematisch adressiert und TYPO3 zu einem KI-fähigen CMS weiterentwickelt[6].

Die Nutzung von KI in TYPO3 ist aber schon jetzt nicht mehr nur hypothetisch, sondern teilweise bereits in ersten einfachen Varianten umgesetzt worden. So ermöglichen bestehende TYPO3-Erweiterungen bereits heute die Integration von Large Language Models (LLMs) wie ChatGPT, um etwa Inhalte automatisch zu generieren oder zu übersetzen sowie SEO-Analysen durchzuführen[7] [8]. Auch Medieninhalte wie Bilder lassen sich mithilfe KI-gestützter Funktionen direkt im Backend erzeugen[9].

Ein konkretes Beispiel ist die Extension T3AI, die verschiedene KI-Funktionen wie Textgenerierung, Übersetzung, Bildkreation und SEO-Optimierung innerhalb einer einheitlichen Benutzeroberfläche zusammenführt[10]. Solche Integrationen helfen Redakteur*innen, wiederkehrende Aufgaben effizienter zu bewältigen und gleichzeitig die inhaltliche Konsistenz zu wahren. Strategisch betrachtet reagiert TYPO3 damit auf den zunehmenden Innovationsdruck durch generative KI. Statt den CMS-Kern grundlegend zu verändern, wird ein modularer Weg über Erweiterungen und APIs eingeschlagen, um flexible und nutzerzentrierte Lösungen zu ermöglichen[11].

[6] Nägler, „TYPO3 V14 – AI Integrations“.
[7] Manuel Schnabel, „Extension AI SEO-Helper — AI SEO-Helper 0.7 Documentation“, 2024, https://docs.typo3.org/p/passionweb/ai-seo-helper/0.7/en-us/ letzter Zugriff 12.09.2025 12:53 Uhr.
[8] NITSAN Technologies, „T3AI: All-in-One TYPO3 AI Extension“, 1. Juli 2025, https://github.com/nitsan-technologies/ns_t3ai?tab=readme-ov-file letzter Zugriff 12.09.2025 13:00 Uhr.
[9] DMK E-BUSINESS GmbH, *typo3-mkcontentai*, PHP, 17. Mai 2023; DMK E-BUSINESS GmbH, released 28. Juli 2025, https://github.com/DMKEBUSINESSGMBH/typo3-mkcontentai letzter Zugriff 12.09.2025 12:55 Uhr.
[10] NITSAN Technologies, „T3AI (Ns_t3ai)“.
[11] Nägler, „TYPO3 V14 – AI Integrations“.

2.2 Semantische Datenformate

Strukturierte und maschinenlesbare Inhalte im Web basieren auf einem Graphmodell, das Beziehungen zwischen Ressourcen explizit beschreibt. Das Resource Description Framework in der Version 1.1 definiert dafür ein Datenmodell aus Tripeln aus Subjekt, Prädikat und Objekt sowie die Zusammenfassung solcher Tripel in Graphen und Datasets.[12] Diese Abstraktion erlaubt die eindeutige Referenzierung von Entitäten, Eigenschaften und Relationen und schafft die Grundlage für interoperablen Datenaustausch über Systemgrenzen hinweg.[13]

Für Web und API ist eine JSON-basierte Serialisierung zentral. JSON-LD 1.1 überführt RDF-Graphen in reguläres JSON und erlaubt über den Context die Abbildung verwendeter Begriffe auf definierte Vokabulare.[14] So lassen sich bestehende JSON-Workflows schrittweise semantisieren, ohne dass Entwicklungsteams das gesamte RDF-Vokabular sofort beherrschen müssen.[15] Das zugrunde liegende JSON-Format ist in RFC 8259 normiert und beschreibt die elementaren Datentypen sowie die Syntax, auf die sich moderne Web-APIs, Logstrukturen und Austauschformate stützen.[16]

Das sichtbare Web wird weiterhin durch HTML beschrieben. Der HTML Living Standard legt Semantik und Verarbeitung von Dokumenten fest und dient als Referenz für die korrekte Einbettung strukturierter Daten in Webseiten.[17] Wo Inhalte unmittelbar im Frontend ausgezeichnet werden, etwa für Suchmaschinen oder assistive Technologien, bildet HTML den verbindlichen Rahmen für semantische Markup-Varianten und Einbettungsformen.

Ein verbreitetes Vokabular zur Beschreibung von Entitäten ist schema.org. Es stellt Typen und Eigenschaften für häufige Domänen wie Personen, Organisationen, Orte, Ereignisse oder Creative Works bereit und kann in RDFa,

---

[12] World Wide Web Consortium (W3C), „RDF 1.1 Concepts and Abstract Syntax“, W3C Technical Reports, 25. Februar 2014, https://www.w3.org/TR/rdf11-concepts/.
[13] World Wide Web Consortium (W3C), „W3C, RDF 1.1 Concepts“.
[14] Manu Sporny u. a., „JSON-LD 1.1“, W3C Technical Reports, 16. Juli 2020, https://www.w3.org/TR/json-ld11/.
[15] Sporny u. a., „W3C, JSON-LD 1.1“.
[16] Tim Bray, *The JavaScript Object Notation (JSON) Data Interchange Format*, RFC 8259, RFC Editor, Internet Standard (STD 90) Dezember 2017, https://doi.org/10.17487/RFC8259.
[17] *HTML Standard*, Spezifikation, Version Living Standard, WHATWG, living standard, https://html.spec.whatwg.org/ zuletzt geprüft am 11.09.2025 18:49 Uhr.

Microdata oder JSON-LD eingebunden werden.[18] In dieser Arbeit ist schema.org vor allem relevant, wenn Inhalte für nachgelagerte Systeme und Retrieval-Prozesse mit minimalem, aber präzisem Kontext versehen werden.[19]

Für die maschinelle Interaktion zwischen Komponenten wird häufig ein leichtgewichtiges RPC-Paradigma genutzt. JSON-RPC 2.0 definiert Anfragen und Antworten als JSON-Objekte mit Methodennamen, Parametern und Korrelation via id. Das Protokoll ist transportunabhängig und eignet sich damit für HTTP, WebSockets oder lokale Kanäle.[20] In den Architekturteilen dieser Arbeit erleichtert diese Spezifikation die saubere Trennung von Lese- und Schreibpfaden sowie die Protokollierung.

Sobald Schnittstellen schreibend auf geschützte Ressourcen zugreifen, sind standardisierte Autorisierungs- und Identitätsflüsse erforderlich. OAuth 2.0 spezifiziert ein Framework, mit dem ein Client nach Einwilligung des Ressourceneigentümers ein Zugriffstoken erhält und damit begrenzte Rechte gegenüber einer API ausübt.[21] Aufbauend darauf liefert OpenID Connect eine Identitätsschicht, mit der Clients die Identität eines Endnutzers verifizieren und standardisierte Claims beziehen können.[22] Diese Kombination ist in dieser Arbeit relevant, um klare Verantwortlichkeiten, least-privilege-Zugriffe und nachvollziehbare Audit-Trails umzusetzen.

Zusammengefasst stellen diese Standards die Grundlage für die weiteren Kapitel dar. RDF liefert das semantische Modell, JSON-LD die praktikable Serialisierung in JSON, HTML den Dokumentrahmen für eingebettete Strukturen, schema.org ein etabliertes Vokabular, JSON-RPC 2.0 das einfache Nachrichtenformat für maschinelle Interaktion und OAuth 2.0 mit OpenID Connect die abgesicherten Zugriffs- und Identitätsflüsse.

---

[18] Schema.org, „Schema.org“, zugegriffen 11. September 2025, https://schema.org/ zuletzt geprüft am 11.09.2025 19:03 Uhr.

[19] Schema.org, „schema.org“.

[20] JSON-RPC Working Group, „JSON-RPC 2.0 Specification“, Jsonrpc.Org, 1. Juli 2010, https://www.jsonrpc.org/specification zuletzt geprüft am 11.09.2025 19:15 Uhr.

[21] Dick Hardt, *The OAuth 2.0 Authorization Framework*, Request for Comments RFC 6749 (Internet Engineering Task Force, 2012), https://doi.org/10.17487/RFC6749 zuletzt geprüft am 11.09.2025 19:33 Uhr.

[22] OpenID Foundation, „OpenID Connect Core 1.0“, Openid.Net, 15. Dezember 2023, https://openid.net/specs/openid-connect-core-1_0.html letzter Zugriff 11.09.2025 19:46 Uhr.

2.3 LLMs und Retrieval-Augmented Generation (RAG)

Large Language Models, kurz LLMs, sind eine fundamentale technologische Entwicklung im Bereich der Künstlichen Intelligenz. Insbesondere im Bereich der Verarbeitung natürlicher Sprache (Natural Language Processing, NLP). Sie basieren in der Regel auf sogenannten Transformer-Architekturen, die erstmals im Jahr 2017 von Vaswani et al. im wegweisenden Paper "Attention Is All You Need" vorgestellt wurden[23]. Der Transformer unterscheidet sich von vorherigen neuronalen Architekturen insbesondere durch das sogenannte Self-Attention-Prinzip, welches es erlaubt, Beziehungen zwischen allen Tokens eines Textes gleichzeitig zu berechnen. Diese parallele Verarbeitung ermöglicht eine effizientere Kontextualisierung von Sprache und ist ein wesentlicher Grund dafür, dass Transformer-Modelle heute den Standard in der Sprachverarbeitung darstellen.

Large Language Models wie GPT, Claude oder Gemini nutzen dieses Prinzip und erweitern es um gewaltige Mengen an Parametern und Trainingsdaten. Frühere Modelle wie GPT-3 oder PaLM legten die Basis, während aktuelle Generationen durch deutlich verbesserte Kontextverarbeitung, geringere Fehleranfälligkeit und multimodale Fähigkeiten (Text, Bild, Audio) neue Maßstäbe setzen und diese Entwicklung durch gewaltige Mengen an Parametern und Trainingsdaten weiter ausbauen. Diese Modelle werden zunächst im Rahmen eines selbstüberwachten Lernverfahrens auf gigantischen Sammlungen aus Textdaten (sogenannten Textkorpora) vortrainiert[24]. Dabei lernen sie statistische Zusammenhänge zwischen Wörtern, Sätzen und Textstrukturen kennen. Anschließend erfolgt in vielen Fällen ein sogenanntes Fine-Tuning, bei dem das Modell für spezifische Aufgaben oder Domänen weiter angepasst wird[25]. Das Ergebnis sind KI-Systeme, die in der Lage sind, menschliche Sprache mit beeindruckender Kohärenz,

[23] Ashish Vaswani u. a., „Attention Is All You Need“, arXiv:1706.03762, preprint, arXiv, 2. August 2023, https://doi.org/10.48550/arXiv.1706.03762 letzter Zugriff 11.09.2025 22:09 Uhr.
[24] Tom B. Brown u. a., „Language Models are Few-Shot Learners“, arXiv:2005.14165, preprint, arXiv, 22. Juli 2020, https://doi.org/10.48550/arXiv.2005.14165 letzter Zugriff 11.09.2025 22:10 Uhr.
[25] Colin Raffel u. a., „Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer“, arXiv:1910.10683, Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer, preprint, arXiv, 19. September 2023, https://doi.org/10.48550/arXiv.1910.10683 letzter Zugriff 11.09.2025 22:13 Uhr.

Kontextsensitivität und grammatikalischer Präzision zu verarbeiten und zu erzeugen.

Dennoch bleiben Herausforderungen bestehen. Eines der zentralen Probleme ist das sogenannte "Halluzinieren". LLMs generieren Inhalte, die sprachlich korrekt und plausibel erscheinen, aber faktisch falsch oder frei erfunden sind. In dieser Arbeit wird dabei insbesondere auf die Systematisierung und Definition aus Ji et al. (2023) zurückgegriffen, wobei sich der Bezug vor allem auf Kapitel 3 der Publikation konzentriert, in dem verschiedene Typen von Halluzinationen in der natürlichen Sprachgenerierung beschrieben werden[26]. Dieses Verhalten resultiert aus der rein statistischen Natur der Sprachgenerierung, denn das Modell entscheidet sich für wahrscheinlich klingende Wortfolgen, unabhängig davon, ob die darin enthaltenen Informationen korrekt oder überprüfbar sind. In einem öffentlichen oder behördlichen Kontext, in dem Vertrauenswürdigkeit und Nachvollziehbarkeit zentral sind, stellt dies ein signifikantes Problem dar.

Um dieses Problem zu adressieren, wurde das Konzept der Retrieval-Augmented Generation, kurz RAG, entwickelt. Dieses wurde unter anderem von Lewis et al. (2020) geprägt, wobei sich die Aussagen in dieser Arbeit insbesondere auf die Kapitel 2 und 4 der Publikation beziehen[27] und stellt einen Paradigmenwechsel dar. Anstatt ausschließlich auf das in den Modellparametern gespeicherte Wissen zuzugreifen, werden zusätzlich externe Dokumentenquellen eingebunden. Ein RAG-System besteht aus zwei Hauptkomponenten: einem Retriever und einem Generator. Der Retriever durchsucht einen externen Wissensspeicher, typischerweise in Form einer Vektordatenbank, nach relevanten Dokumenten oder Textabschnitten. Diese Datenbank enthält zuvor aufbereitete Inhalte, die durch sogenannte Embeddings, also semantische Vektorrepräsentationen von Texten, durchsucht werden können. Die Ausführungen in dieser Arbeit stützen sich dabei auf die methodischen Details in Kapitel 3 des Papers von Reimers und Gurevych (2019)[28]. Die am besten passenden Inhalte werden vom Retriever ausgewählt und dem LLM als zusätzlicher Kontext zur Verfügung gestellt. Das

[26] Ziwei Ji u. a., „Survey of Hallucination in Natural Language Generation“, *ACM Computing Surveys* 55, Nr. 12 (2023): 1–38, https://doi.org/10.1145/3571730 letzter Zugriff 11.09.2025 22:19 Uhr.

[27] Patrick Lewis u. a., „Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks“, arXiv:2005.11401, preprint, arXiv, 12. April 2021, https://doi.org/10.48550/arXiv.2005.11401 letzter Zugriff 11.09.2025 22:22 Uhr.

[28] Nils Reimers und Iryna Gurevych, „Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks“, arXiv:1908.10084, preprint, arXiv, 27. August 2019, https://doi.org/10.48550/arXiv.1908.10084 letzter Zugriff 11.09.2025 22:23 Uhr.

LLM generiert dann eine Antwort, die nicht nur auf seinem internen Wissen basiert, sondern auch die abgerufenen Informationen berücksichtigt. Dadurch wird die Genauigkeit der generierten Antworten signifikant erhöht und zugleich können Quellen explizit benannt werden, was die Transparenz verbessert. In dieser Arbeit wird hierbei vor allem auf Kapitel 5 der Publikation von Izacard und Grave Bezug genommen[29].

Ein entscheidender Vorteil von RAG besteht darin, dass der Wissensspeicher unabhängig vom LLM aktualisiert werden kann. Neue Informationen, etwa geänderte Gesetzestexte, aktuelle Verwaltungsvorgaben oder neue Forschungsergebnisse, lassen sich einfach in die Vektordatenbank einpflegen, ohne dass das Sprachmodell selbst neu trainiert werden muss. Die technische Umsetzung und Vorteile für Aktualisierungsprozesse werden in Kapitel 4 des Papers von Karpukhin et al. (2020) thematisiert[30]. Dies macht RAG besonders attraktiv für Szenarien, in denen Informationsstände sich häufig ändern oder hohe Anforderungen an Nachvollziehbarkeit und Aktualität gestellt werden. Darüber hinaus erlaubt der Einsatz externer Dokumente die Verwendung deutlich kleinerer Modelle, da nicht alle Informationen in den Modellparametern enthalten sein müssen. Dies reduziert den Rechenaufwand und macht den Einsatz solcher Systeme auch für Organisationen mit begrenzten Ressourcen praktikabel. Die zugrundeliegenden Prinzipien für die Kombination von Retrieval und Sprachmodelltraining werden dabei vor allem in Kapitel 3 des Papers von Guu et al. (2020) beschrieben[31].

Im Kontext dieser Arbeit ist Retrieval-Augmented Generation besonders relevant, da es eine Brücke zwischen der Welt der generativen KI und der strukturierten Informationsbereitstellung in Content-Management-Systemen schlägt. TYPO3, als weit verbreitetes CMS im öffentlichen Sektor, bietet durch seine modulare Architektur und die Möglichkeit, Inhalte in semantischen Formaten wie JSON-LD oder über strukturierte Datenmodelle bereitzustellen, eine ideale Grundlage für

[29] Gautier Izacard und Edouard Grave, „Leveraging Passage Retrieval with Generative Models for Open Domain Question Answering“, arXiv:2007.01282, preprint, arXiv, 3. Februar 2021, https://doi.org/10.48550/arXiv.2007.01282 letzter Zugriff 12.09.2025 09:47.

[30] Vladimir Karpukhin u. a., „Dense Passage Retrieval for Open-Domain Question Answering“, in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, hg. von Bonnie Webber u. a. (Association for Computational Linguistics, 2020), https://doi.org/10.18653/v1/2020.emnlp-main.550 letzter Zugriff 12.09.2025 09:48 Uhr.

[31] Kelvin Guu u. a., „REALM: Retrieval-Augmented Language Model Pre-Training“, arXiv:2002.08909, preprint, arXiv, 10. Februar 2020, https://doi.org/10.48550/arXiv.2002.08909 letzter Zugriff 12.09.2025 09:50 Uhr.

RAG-Anwendungen. Insbesondere in Verbindung mit dem Model Context Protocol (MCP), welches auf die maschinenlesbare, kontextuell strukturierte Publikation von Inhalten abzielt, ergibt sich ein synergetisches Potenzial. Inhalte, die über TYPO3 erstellt und gepflegt werden, können so semantisch angereichert und in einer Vektordatenbank indexiert werden. Dadurch wird es möglich, diese Informationen über ein RAG-System für generative KI-Anwendungen zugänglich zu machen, etwa in Form von interaktiven Bürgerportalen, FAQ-Chatbots oder internen Assistenzsystemen für Verwaltungsmitarbeitende.
Diese Verbindung eröffnet neue Wege für eine vertrauenswürdige, nachvollziehbare und aktuelle Informationsbereitstellung, bei der Nutzer*innen nicht nur eine generierte Antwort erhalten, sondern auch direkt auf die zugrunde liegende Quelle zugreifen können. In einem Umfeld, das zunehmend von der Integration intelligenter Systeme geprägt ist, stellt RAG somit eine zentrale Technologie dar, um die Schnittstelle zwischen maschineller Sprachverarbeitung und redaktionell gepflegten Inhalten zukunftssicher zu gestalten.

2.4 Model Context Protocol (MCP)

„Think of MCP like a USB-C port for AI applications." [32]
Mit diesem Bild wird von der offiziellen Dokumentation des Model Context Protocol (MCP) die zentrale Funktion dargestellt. Im Zuge der zunehmenden Integration generativer KI-Systeme in redaktionelle und verwaltungsbezogene Prozesse gewinnt die Frage an Relevanz, wie maschinenlesbare Inhalte modular, standardisiert und sicher bereitgestellt werden können. Das Model Context Protocol (MCP), im November 2024 von Anthropic vorgestellt, fungiert als universeller Vermittlungsmechanismus, der strukturierte Kontexte für KI-Anwendungen über standardisierte Schnittstellen zugänglich macht[33]. Ziel des Protokolls ist es, externe Wissensquellen wie Datenbankabfragen, API-Ergebnisse oder dynamische CMS-Inhalte so zu kapseln, dass sie systematisch in die Antwortgenerierung von LLMs eingebunden werden können.

[32] Anthropic, „What Is the Model Context Protocol (MCP)?", Model Context Protocol, zugegriffen 12. September 2025, https://modelcontextprotocol.io/docs/getting-started/intro letzter Zugriff 12.09.2025 09:53 Uhr.
[33] Anthropic, „Introducing the Model Context Protocol", Anthropic News, 25. November 2024, https://www.anthropic.com/news/model-context-protocol letzter Zugriff 12.09.2025 09:55 Uhr.

MCP basiert auf dem JSON-RPC-2.0-Standard als Transportprotokoll, wobei sogenannte „Tools" registriert werden können, die durch das LLM bei Bedarf dynamisch aktiviert werden. Diese Tools stellen Funktionsmodule dar, die auf konkrete Daten oder Dienste zugreifen. Die semantische Struktur der Tool-Beschreibungen erlaubt es, KI-Systeme in einer kontrollierten Umgebung kontextsensitiv zu erweitern. Ein zentraler Mechanismus ist dabei die kontextuelle Aktivierung. Nur relevante Tools werden vom Modell verwendet, wodurch sowohl Effizienz als auch Sicherheit verbessert werden[34].

Diese Architektur unterscheidet sich von klassischen Prompt- oder Retrieval-Ansätzen insofern, als sie deklarativ und validierbar bleibt. Alle Tool-Aufrufe erfolgen auf Basis eines expliziten, versionierten Interfaces. Dadurch bleibt der Kontext jederzeit nachvollziehbar. Dies entspricht insbesondere den Anforderungen des öffentlichen Sektors hinsichtlich Transparenz, Kontrolle und Informationshoheit[35].

In einer aktuellen Überblicksstudie heben Hou et al. (2025) hervor, dass MCP durch seine Trennung von Modell- und Kontextlogik sowohl sicherheitstechnisch als auch architektonisch ein Fortschritt gegenüber bisherigen Lösungen darstellt[36]. Besonders relevant ist hierbei die Fähigkeit, maschinengenerierte Inhalte mit klaren Herkunfts- und Funktionspfaden zu verknüpfen. Dies eröffnet neue Perspektiven für den Einsatz in regulierten Umgebungen wie zum Beispiel Ministeriumswebseiten oder Klinikportalen.

Erste Implementierungen im Drupal-Umfeld zeigen, wie redaktionelle Inhalte mithilfe des MCP strukturiert bereitgestellt und KI-gesteuert verarbeitet werden können. Für TYPO3 fehlen bisher entsprechende Ansätze. Aus diesem Grund widmet sich die vorliegende Arbeit dem Ziel, ein tragfähiges konzeptionelles Modell zur MCP-Integration in TYPO3 zu entwickeln.

---

[34] Anthropic, „Model Context Protocol Specification", Model Context Protocol, 18. Juni 2025, https://modelcontextprotocol.io/specification/2025-06-18 letzter Zugriff 12.09.2025 10:03 Uhr.
[35] Anthropic, „MCP Spec v0.2.0".
[36] Xinyi Hou u. a., „Model Context Protocol (MCP): Landscape, Security Threats, and Future Research Directions", arXiv:2503.23278, preprint, arXiv, 6. April 2025, https://doi.org/10.48550/arXiv.2503.23278 letzter Zugriff 12.09.2025 10:10 Uhr.

2.5 Methodischer Rahmen der Konzeptentwicklung

Dieses Kapitel beschreibt das Forschungsdesign der empirischen Erhebung. Ziel der Erhebung ist es, die in Kapitel 2 und 2.5 hergeleiteten Annahmen zur MCP-Integration in TYPO3 aus Praxissicht zu prüfen und zu priorisieren. Dafür wird eine qualitative Vorgehensweise mit standardisiertem Fragebogen gewählt, die strukturierte Antworten erlaubt und dennoch Raum für offene Begründungen lässt.

Die Stichprobe umfasst Fachpersonen aus TYPO3-, CMS- und KI-Kontext. Das Instrument erfasst Vorerfahrung, erwarteten Nutzen, Anforderungen im öffentlichen Umfeld sowie Einschätzungen zu Workflows, Rollen und Rechten, Versionierung, Logging und Schnittstellenstandards. Die Durchführung erfolgt schriftlich und einheitlich. Die Auswertung kombiniert deskriptive Berichte geschlossener Items mit einer thematischen Bündelung der Freitextanteile. Die Befunde werden anschließend den in Kapitel 5 operationalisierten Kontrollpunkten zugeordnet und dienen dort als Grundlage für die Bewertung der Konzeption. Diese theoretischen Vorarbeiten werden durch qualitative Expertenbefragung ergänzt, die in Kapitel 3 methodisch eingeordnet werden. Ziel dieser empirischen Phase ist es, die Machbarkeit, Relevanz und Akzeptanz der MCP-Integration in TYPO3 aus Sicht von Fachpersonen aus dem CMS-, KI- und Verwaltungskontext zu bewerten. Die Ergebnisse der Befragung dienen nicht als eigenständiger Datenbestand, sondern als Validierungsschicht der konzeptionellen Überlegungen. Die dabei gewonnenen Erkenntnisse fließen direkt in die Ausgestaltung von Systemarchitektur, Schnittstellendesign und Umsetzungsszenarien ein.

Die Entscheidung für diesen methodischen Rahmen ergibt sich aus der spezifischen Forschungsfrage und dem Anwendungsfokus. Die Arbeit bewegt sich im Spannungsfeld zwischen technologischer Innovation und öffentlicher Infrastruktur, weshalb eine isolierte technische Betrachtung nicht zielführend wäre. Vielmehr wird ein ganzheitlicher Zugriff angestrebt, der sowohl die strukturellen Voraussetzungen von TYPO3 als auch die Anforderungen vertrauenswürdiger KI-Systeme im öffentlichen Raum berücksichtigt.

3. Methodik

3.1 Forschungsdesign

Die vorliegende Arbeit verfolgt einen konzeptionellen Forschungsansatz mit qualitativer Ausrichtung. Ziel ist die Entwicklung eines Architekturmodells zur Integration des Model Context Protocols (MCP) in TYPO3. Der Fokus liegt auf Anforderungen des öffentlichen Sektors sowie auf fachlichen, technischen und governancebezogenen Voraussetzungen in TYPO3 (Workflows, Rollen- und Rechtemodelle, Versionierung, Logging, Schnittstellen- und Tool-Verträge). Im Mittelpunkt steht keine unmittelbare Implementierung. Stattdessen soll durch theoretische Modellbildung und durch praxisnahe Einschätzungen von Expertinnen und Experten ein Konzept entstehen, das sich auf andere Kontexte übertragen lässt. Methodisch basiert die Arbeit auf einer Kombination aus strukturierter Literatur- und Dokumentenanalyse sowie qualitativen Expertenbefragungen. Die Auswertung der Freitextanteile erfolgte themengeleitet. Aus den Forschungsfragen und dem theoretischen Rahmen (Kapitel 2) wurden zentrale Themenbereiche vorab definiert (u. a. Technik/Architektur, Sicherheit/Datenschutz, Governance/Human-in-the-Loop, Redaktionsassistenz, RAG/Suche) und die Antworten entlang dieser Bereiche zusammengefasst. Wo im Material neue, relevante Aspekte auftraten, wurden die Themenbereiche pragmatisch erweitert. Die Ergebnisse werden je Themenbereich verdichtet dargestellt und durch kurze Originalzitate illustriert. Eine vollumfängliche, regelgeleitete Inhaltsanalyse wurde nicht durchgeführt.

3.2 Auswahl der Befragten

Die Befragten wurden gezielt ausgewählt, damit relevante Perspektiven vertreten sind. Eingeladen wurden Personen aus dem TYPO3, CMS und KI-Umfeld, die entweder zur Weiterentwicklung von TYPO3 beitragen, Projekte mit Behörden kennen oder besondere Erfahrung mit KI und semantischen Technologien mitbringen. Ziel war es, ein möglichst breites Spektrum an Perspektiven abzubilden, welches sich von Core-Entwicklern über Agenturverantwortliche bis hin zu KI-Strateginnen erstreckt. Die Auswahlkriterien umfassten:

- Relevante Expertise im TYPO3- oder CMS-Ökosystem
- Technisches oder strategisches Wissen im Bereich Künstliche Intelligenz
- Bezug zu semantischen Technologien oder öffentlichkeitsrelevanten CMS-Projekten

Am Ende der Umfrage konnten die Befragten angeben, ob sie in dieser Arbeit namentlich erwähnt werden möchten. Wurde dies abgelehnt, erfolgt eine anonyme Darstellung. Die nachfolgende Übersicht stellt die befragten Personen und ihre fachliche Einordnung dar:

- Frank Nägler – CTO TYPO3 GmbH, Initiator des MCP-Channels auf Slack
- Alexander Bernhardt – Gründer von Hauptsache.net, CMS- und TYPO3-Spezialist
- Thomas Schöne – Senior PHP-Entwickler bei Netresearch DTT GmbH
- Olivier Dobberkau – Präsident der TYPO3 Association, CEO der dkd Internet GmbH
- Oli Bartsch – TYPO3 Core Developer
- Marco Pfeiffer – Entwickler einer MCP-TYPO3-Extension, aktiv im Core-AI-Slack-Kanal und Developer bei Hauptsache.net
- Rico Loschke – AI-Trainer und AI-Artist mit CMS-Erfahrung
- Florian Froidevaux, Marketing Manager bei in2code, mit Fokus auf Marketing Analytics im TYPO3 Ökosystem und Co Team Lead im Marketing Team der TYPO3 Association
- Thomas Esders – Co-Founder von Leuchtfeuer, TYPO3-basierte Agenturlösungen
- Manuel Schnabel und André Kraus, Gründer der AutoDudes GbR und verantwortlich für die TYPO3 AI Suite

Die Auswahl stellt sicher, dass sowohl technische als auch strategische Einschätzungen aus der Praxis abgebildet werden.

3.3 Fragebogenerstellung und Durchführung

3.3.1 Ziel und Aufbau des Fragebogens

Der Fragebogen soll zwei Dinge leisten. Erstens klärt er, wie die Befragten die Rolle und den möglichen Nutzen von KI im CMS Kontext einschätzen. Zweitens sammelt er Anforderungen und Empfehlungen, die für ein MCP basiertes Integrationskonzept relevant sind. Das Instrument ist strukturiert aufgebaut und kombiniert feste Fragen mit offenen Textfeldern. Abgedeckt werden Vorerfahrung, erwartete Potenziale von KI im CMS, Hürden und Anforderungen im öffentlichen Umfeld, die Einschätzung zu einem Standard wie MCP sowie Empfehlungen zu Komponenten und Architektur. Der vollständige Wortlaut des Instruments, die Einleitungstexte und der Hinweis zur namentlichen Nennung sind im Anhang dokumentiert.

3.3.2 Fragetypen und Begründung der Messlogik

Je nach Frageform werden passende Auswertungen vorgenommen.
a) Bei Ja- oder Nein-Fragen wird der Anteil der Zustimmungen berichtet. Diese Kennzahl ist verständlich, vergleicht Gruppen einfach und lässt sich später direkt in Kapitel 5 einordnen.
b) Offene Antworten werden thematisch zusammengefasst und mit kurzen Belegzitaten dokumentiert. So lassen sich wiederkehrende Muster herausarbeiten, die konkrete Architekturentscheidungen stützen.
c) Falls ordinale Skalen eingesetzt werden, werden Häufigkeiten je Kategorie sowie Median und Interquartilsabstand berichtet. Diese Maße passen zu Skalen mit geordneter Abstufung.

3.3.3 Durchführung und Dokumentation

Die Befragung wurde schriftlich über Google Forms durchgeführt. Alle Befragten erhielten identische Fragen in derselben Reihenfolge. Die Teilnahme kann anonym erfolgen. E-Mail-Adressen dienen ausschließlich der Kommunikation. Eine namentliche Nennung erfolgt nur mit ausdrücklicher Zustimmung. Die

Antworten werden in aggregierter Form ausgewertet und in Kapitel 4 berichtet. Die genaue Fassung des Fragebogens sowie die Hinweise zur Nennung sind im Anhang hinterlegt.

3.3.4 Auswertungsschritte

Die Auswertung verläuft in drei Schritten. Zuerst werden die geschlossenen Fragen deskriptiv berichtet. Danach werden die offenen Antworten thematisch codiert, indem wiederkehrende Aspekte in prägnante Themen gebündelt und mit Belegzitaten unterlegt werden. Abschließend werden die Befunde mit den Konzepten aus Kapitel 2 und den Architekturzielen in Kapitel 5 verknüpft. So entsteht eine nachvollziehbare Brücke zwischen Praxisfeedback und den vorgeschlagenen Sicherheitsankern, Workflows und Komponenten.

3.4 Grenzen und Limitationen der Methode

Die gewählte Methodik ermöglicht eine fundierte Analyse von Einschätzungen und Anforderungen im praktischen Umfeld. Dennoch sind bestimmte Einschränkungen zu beachten. Die geringe Anzahl an Befragten erlaubt keine quantitativen Verallgemeinerungen. Ziel ist vielmehr die theoriegeleitete Konzeptvalidierung mit Rückgriff auf fundierte Fachmeinungen.
Eine weitere Begrenzung ergibt sich aus dem gewählten Online-Format. Rückfragen oder spontane Nachfragen während der schriftlichen Expertenbefragung waren nicht möglich. Dafür bietet die schriftliche Erhebung eine hohe Vergleichbarkeit und Transparenz der Ergebnisse. Die Auswahl der Experten ist zudem auf das TYPO3-Umfeld fokussiert. Übertragungen auf andere CMS-Systeme sind daher nur eingeschränkt möglich.

4. Ergebnisse

4.1 Ergebnisse der Befragung

Die schriftliche Expertenbefragung umfasste zehn Personen aus dem TYPO3-, CMS- und KI-Umfeld und lief vom 21.06.2025 bis 26.08.2025. Alle verfügen über einschlägige Vorerfahrungen. Die Auswertung folgt dem in Kapitel 3 beschriebenen Vorgehen.
Die Mehrheit der Befragten arbeitet in Agenturen und der CMS-Entwicklung, einzelne Nennungen entfallen auf Beratung und die TYPO3 GmbH. Alle Befragten gaben an, bereits Erfahrungen mit TYPO3, anderen CMS oder KI-basierten Systemen gesammelt zu haben. Alle beurteilten den MCP-Ansatz als grundsätzlich sinnvoll. Die namentliche Nennung wurde vollständig freigegeben. Diese Ergebnisse entsprechen der Zielsetzung der Erhebung, praxisnahe Einschätzungen aus dem TYPO3-Ökosystem für die Konzeptvalidierung zu gewinnen. Die Auswertung folgt dem in Kapitel 3 erläuterten Vorgehen mit deskriptiver Berichterstattung der geschlossenen Fragen und thematischer Bündelung der offenen Antworten.

Thematische Schwerpunkte der offenen Antworten lassen sich wie folgt verdichten:

- Technik und Architektur wurden am häufigsten adressiert. Genannt wurden Schnittstellen, API-Design, MCP-Tools, Ressourcen und Prompts, Authentifizierung und Betriebsfragen.
- Sicherheit und Datenschutz betonen die Befragten regelmäßig. Wiederkehrend sind Anforderungen wie DSGVO-Konformität, Datenminimierung, klare Sichtbarkeitsgrenzen, EU-Hosting und Vermeidung von Abhängigkeiten.
- Governance und Human-in-the-Loop werden als Voraussetzung gesehen. Erwartet werden Freigabeprozesse, Protokollierung, nachvollziehbare Änderungen und kontrollierte Schreibwege.
- Redaktionsassistenz und Automatisierung werden als Potenzial gesehen, etwa für Übersetzungen, Alternativtexte, Vereinfachungen, SEO-Hilfen und Metadatenvorschläge.

- RAG und Suche werden als Anschlussoption für Wissensbereitstellung genannt, sofern kontextuelle Daten sauber vorbereitet werden.

4.2 Relevanz, Erwartungen und Herausforderungen

4.2.1 Relevanz und Nutzen

Die Befragten bewerten MCP übereinstimmend als einen zweckmäßigen Standard, um KI-Funktionen kontrolliert an ein CMS anzubinden. Im Mittelpunkt steht die Erwartung, dass strukturierte Kontextbereitstellung und eindeutige Verträge zwischen MCP-Client und CMS die Integration vereinheitlichen und damit die spätere Pflege und Weiterentwicklung erleichtern. Als zentraler Nutzen wird die Möglichkeit genannt, Dienste auszutauschen, ohne die bestehende CMS-Logik grundlegend zu verändern. Das betrifft sowohl Modellanbieter als auch spezialisierte Werkzeuge, die über MCP als Tools angebunden werden.

> „Arbeitserleichterung für Redakteure. Text Generierung oder Umschichtung von Content etc."
> Marco Pfeiffer [37]

Auf redaktioneller Ebene werden wiederkehrende Assistenzaufgaben hervorgehoben. Genannt werden Übersetzungen, Alternativtexte, Zusammenfassungen, konsistente Metadatenvorschläge sowie Hilfen beim Formulieren und Strukturieren. Solche Unterstützungen sollen die Einarbeitung in Backends erleichtern und Arbeitsabläufe beschleunigen, ohne die Hoheit über die Inhalte aufzugeben. Für Entwicklungsteams sind neben der Austauschbarkeit insbesondere klare Schnittstellenverträge bedeutsam, etwa wohldefinierte Tool- und Resource-Schemata, Versionierbarkeit und reproduzierbares Testen. Die Kombination aus redaktionellem Effizienzgewinn und technischer Standardisierung wird als wesentliche Begründung für die Relevanz von MCP genannt.

[37] Eigene Erhebung (Expertenumfrage), Marco Pfeiffer, Antwort auf Frage 4

„[…] Bei der Personalisierung sehr großer Nutzen. Verklausulierte Verordnungen/Formulare/Anträge etc. Erklärungen können durch KI viel genauer zum Nutzer angepasst werden. Hilfe bei der Bearbeitung dieser, etc."
Florian Froidevaux [38]

4.2.2 Anforderungen im öffentlichen Kontext

Für den öffentlichen Kontext werden Anforderungen benannt, die sowohl regulatorische als auch betriebliche Aspekte betreffen. An erster Stelle stehen Datenschutz nach DSGVO, transparente Verarbeitung, nachvollziehbare Datenflüsse und klare Verantwortlichkeiten.

„Datenschutz, Privacy, natürlich auch Erweiterbarkeit und andere Standards, aber die Realität ist oft, dass Systeme […] so hingebogen werden, dass von Einfachheit und Standards weniger übrigbleibt."
Rico Loschke. [39]

Die Erwartungen richten sich auf EU-Standorte, auf Datenminimierung bei externen Aufrufen und auf Maßnahmen zur Pseudonymisierung, wo immer dies fachlich vertretbar ist. Die Befragten verknüpfen diese Vorgaben mit dem Anspruch, die Nachvollziehbarkeit für prüfende Stellen sicherzustellen.
Auf CMS-Seite werden Workflows mit Entwurf, Review und Freigabe als verbindliche Prozessanker genannt. Jede KI-Interaktion soll lückenlos protokolliert werden. Gefordert sind granulare Rollen- und Rechtemodelle, die den Grundsatz der minimalen Rechtevergabe unterstützen, sowie eine klare Sichtbarkeit von Kontexten, die an externe Dienste übergeben werden. Für wissensbasierte Assistenz wird außerdem auf semantische Auszeichnung und kuratierte Quellen verwiesen, damit spätere RAG-Szenarien auf verlässlich strukturierte Inhalte zugreifen können. In der Summe entsteht ein Anforderungsbild, das die technische Standardisierung durch MCP mit Governance- und Compliance-Erwartungen des öffentlichen Sektors verzahnt.

[38] Eigene Erhebung (Expertenumfrage), Florian Froidevaux, Antwort auf Frage 3
[39] Eigene Erhebung (Expertenumfrage), Rico Loschke, Antwort auf Frage 9

4.2.3 Umsetzungsherausforderungen

Die Befragten nennen auf technischer Ebene mehrere Hürden. Erstens sind saubere Schnittstellen mit eindeutigem Vertragscharakter erforderlich, einschließlich definierter Schemen für Tools, Prompts und Ressourcen sowie nachvollziehbarer Versionierung. Zweitens werden robuste Authentifizierungs- und Autorisierungskonzepte erwartet, die an vorhandene Rollen- und Rechtekonzepte im CMS anschließen. Drittens werden wohldefinierte Kontexte gefordert, die nur die jeweils erforderlichen Informationen enthalten. Ergänzend werden Vorschau- und Testfunktionen für Prompts, Monitoring für Aufrufe und Quoten sowie aussagekräftiges Logging genannt, um die Betriebsreife zu erhöhen.
Auf organisatorischer Ebene werden Akzeptanz, Qualifizierung und klare Zuständigkeiten betont. Die Befragten empfehlen eine schrittweise Einführung, bei der Schreibvorgänge zunächst als Entwürfe erstellt und anschließend durch menschliche Prüfinstanzen freigegeben werden. Für RAG-Szenarien wird die Qualität der semantischen Aufbereitung als kritischer Erfolgsfaktor beschrieben. Genannt werden konsistente Datenmodelle, belastbare Metadaten und eine transparente Pflege der Wissensquellen. Ohne diese Grundlagen drohen Qualitätsschwankungen und zusätzlicher Prüfaufwand in der Redaktion.

4.2.4 Belegstellen aus der Befragung

Aus den Freitextfeldern lassen sich folgende wiederkehrende Punkte exemplarisch zusammenfassen.

1. Schreibvorgänge sollen zunächst im Entwurfsmodus erfolgen und erst nach menschlicher Freigabe übernommen werden. Die lückenlose Protokollierung gilt als verbindliche Voraussetzung.

2. Gefordert werden EU-Hosting, Datenminimierung und Pseudonymisierung für externe Aufrufe sowie eine Reduktion proprietärer Abhängigkeiten.

3. Für einen Prototyp werden klare MCP-Tool-Verträge, eine Vorschau für Prompts und eine schlanke Integration in bestehende Workflows genannt.

4. Für wissensbasierte Funktionen werden semantische Auszeichnung und kuratierte Quellen als Grundlage für reproduzierbare Ergebnisse eingefordert.

Abschließend ist festzuhalten, dass die Aussagen zur Relevanz, zu den Erwartungen und zu den Herausforderungen konsistent auf ein integratives Bild zulaufen. MCP wird als geeignete Klammer verstanden, um technische Standardisierung und redaktionelle Governance zusammenzuführen. Die genannten Anforderungen und Hürden liefern konkrete Leitplanken für die Ausarbeitung der Systemprinzipien in Kapitel 5.

4.3 Zusammenfassung der empirischen Ergebnisse

Die Befragung mit zehn Experten aus Agenturen, CMS-Entwicklung und Beratung bestätigt den grundsätzlichen Nutzen eines standardisierten Ansatzes zur Anbindung von KI an ein CMS. Alle Befragten verfügen über entsprechende Vorerfahrungen und bewerten den MCP-Ansatz als sinnvoll. Die Antworten verdichten sich zu drei Kernaussagen. Erstens wird ein klarer Vertrag zwischen CMS und Client als Voraussetzung genannt, damit Funktionen austauschbar bleiben und in bestehende Redaktionsprozesse integriert werden können. Zweitens fordern die Befragten für den öffentlichen Kontext Datenschutz nach DSGVO, nachvollziehbare Datenflüsse, EU-Standorte, Transparenz und eine restriktive Rechtevergabe. Drittens werden für die Umsetzung robuste Schnittstellen, wohldefinierte Kontexte, Protokollierung sowie Vorschau und Testmöglichkeiten hervorgehoben.
Aus redaktioneller Sicht erwarten die Befragten vorrangig Entlastung bei wiederkehrenden Aufgaben wie Übersetzungen, Alternativtexten, Zusammenfassungen und Metadatenvorschlägen. Diese Assistenz soll als Entwurf in bestehende Workflows eingebettet und grundsätzlich durch verantwortliche Personen geprüft werden. Für wissensbasierte Funktionen wird die Qualität der zugrunde liegenden Inhalte betont, einschließlich semantischer Auszeichnung und kuratierter Quellen, um reproduzierbare Ergebnisse zu ermöglichen. Für die weitere Arbeit ergeben sich daraus klare Implikationen. In Kapitel 5 sind

Systemprinzipien zu formulieren, die Human-in-the-Loop, minimale Rechtevergabe, lückenlose Nachvollziehbarkeit, Datenminimierung, EU-konforme Betriebsmodelle und offene Schnittstellen in einer konsistenten Architektur zusammenführen. Die Befragung liefert hierfür konkrete Leitplanken: Entwurfswege für Schreiboperationen mit Freigaben, explizite Sichtbarkeit des an extern übergebenen Kontext, Versionierung und Logging, sowie eine Trennung von Zuständigkeiten zwischen Redaktion, Entwicklung und Betrieb. Diese Punkte bilden die Brücke zu den Forschungsfragen und begründen die Ausgestaltung der in Kapitel 5 vorgesehenen Komponenten.
In Bezug auf F1 stützen die Befunde Governance-Mechanismen wie Human-in-the-Loop, Protokollierung, Datenminimierung und EU-Betriebsmodelle. F2 wird getragen durch klare Schnittstellen- und Vertragslogik, wohldefinierte Kontexte sowie Vorschau- und Testmechanismen. F3 wird durch die in der Befragung benannten Voraussetzungen gestützt: Entwurfswege mit nachgelagerter Freigabe, fein abgestuftes Rollen- und Rechtemodell, konsequente Versionierung, lückenlose Protokollierung sowie klar definierte Schnittstellen und Toolverträge. Ergänzend tragen Datenschutz, Transparenz, Barrierefreiheit und Nachvollziehbarkeit zum Gesamtkonzept bei. Zusammen sichern sie Reproduzierbarkeit und Anschlussfähigkeit und bilden die Governancebasis für wissensbasierte Funktionen.

5. Analyse und Bewertung

Dieses Kapitel entwirft eine konsistente Konzeption zur Integration des Model Context Protocols in TYPO3, die die besonderen Anforderungen öffentlicher Einrichtungen adressiert. Ziel ist eine Lösung, die redaktionelle Hoheit und Transparenz sichert, Datenschutz und Barrierefreiheit erfüllt, digitale Souveränität stärkt und strukturell an Retrieval-Augmented-Generation-Verfahren anschließbar ist. Die Konzeption folgt damit den Thesen der Arbeit, wonach KI in Redaktionsprozesse eingebettet, auditierbar und durch menschliche Kontrollinstanzen abgesichert sein muss, während offene Schnittstellen die Anschlussfähigkeit an externe Wissenssysteme gewährleisten. Die normative Grundlage bilden aktuelle Leitlinien und Rechtsakte für den staatlichen KI-Einsatz sowie die MCP-Spezifikation.

Die Expertenbefragung bestätigt dieses Zielbild inhaltlich und legt den Schwerpunkt auf Human-in-the-Loop, lückenlose Nachvollziehbarkeit, minimale Kontextweitergabe und europäische Betriebsmodelle. Die nachfolgenden Abschnitte 5.1 und 5.2 verankern diese Befunde als Gestaltungsziele und Kontrollpunkte der Konzeption, ohne die in Kapitel 4 berichteten Ergebnisse zu wiederholen.[40]

Kapitel 5.1 leitet mit Anforderungen und Use Cases ein und verdichtet die öffentlichen Rahmenbedingungen in konkrete Gestaltungsziele. Kapitel 5.2 überführt diese Ziele in eine Systemarchitektur, die Workspaces, Rechteverwaltung, Versionierung und Logging von TYPO3 nutzbar macht, ohne den Core zu verändern. Die Konzeption setzt auf lose Kopplung über MCP mit versionierten Tool-Verträgen. Dadurch bleiben KI-Dienste austauschbar, was Vendor-Lock-in vermeidet und die digitale Souveränität öffentlicher Stellen stärkt.[41]

[40] Eigene Erhebung, schriftliche Expertenbefragung (Juni–August 2025); vgl. Kap. 4.2–4.3
[41] Anthropic, „MCP Spec v0.2.0“.

5.1 Anforderungen und Use Cases

5.1.1 Gestaltungsprinzipien und rechtliche Rahmenbedingungen

Öffentliche Stellen verlangen für den KI-Einsatz Nachvollziehbarkeit, Transparenz, menschliche Aufsicht und klare Verantwortlichkeiten. Die Leitlinien für den Einsatz Künstlicher Intelligenz in der Bundesverwaltung fordern u. a. Kennzeichnung des KI-Einsatzes, Überprüfbarkeit der Ergebnisse und organisatorische Vorkehrungen, die eine Veröffentlichung ohne menschliche Prüfung ausschließen. Diese Leitplanken sind konzeptionelle Mindestanforderungen an jede MCP-Integration.[42] Ergänzend etabliert der EU Artificial Intelligence Act Querschnittspflichten (u. a. Logging, Transparenz, Human Oversight), die auch für redaktionelle Assistenzsysteme als Best Practice relevant sind.[43] Die Datenschutzkonferenz präzisiert DSGVO-konforme Anforderungen (Zweckbindung, Datenminimierung, Prüfbarkeit auf Richtigkeit/Nicht-Diskriminierung, TOMs) als Voraussetzung für den Betrieb.[44]
Die KI unterstützt nur, sie veröffentlicht nichts selbst. Alle Ergebnisse sind klar gekennzeichnet und gehen erst nach einer menschlichen Prüfung live. Jede Änderung wird versioniert und lückenlos protokolliert.[45] [46] [47]
Die Befragung liefert empirische Evidenz für die zugrunde gelegten Gestaltungsprinzipien. Wiederkehrend gefordert werden ein Prozess mit Entwurf, Review und Veröffentlichung, die Kennzeichnung von Beiträgen der KI, vollständiges Logging sowie klar abgegrenzte Rechte für technische Nutzer*innen. Dies entspricht dem hier verfolgten Ansatz, KI als assistierende Instanz in bestehende Redaktionsprozesse einzubetten.[48]

---

[42] Bundesministerium des Innern, „Leitlinien für den Einsatz Künstlicher Intelligenz in der Bundesverwaltung“, Bundesministerium des Innern, 27. März 2025, https://www.bmi.bund.de/SharedDocs/downloads/DE/publikationen/themen/moderne-verwaltung/ki/BMI25020-leitlinien-ki-bundesverwaltung.pdf letzter Zugriff 13.09.2025 12:27 Uhr.
[43] Europäische Union, „Verordnung (EU) 2024/1689“, 12. Juli 2024, https://eur-lex.europa.eu/eli/reg/2024/1689/oj/eng letzter Zugriff 14.09.2025 12:30 Uhr.
[44] Datenschutzkonferenz (DSK), „Orientierungshilfe ‚KI und Datenschutz‘, Version 1.0“, 6. Mai 2024, https://www.datenschutzkonferenz-online.de/media/oh/20240506_DSK_Orientierungshilfe_KI_und_Datenschutz.pdf letzter Zugriff 14.09.2025 12:37 Uhr.
[45] Bundesministerium des Innern, „BMI, Leitlinien KI Bundesverwaltung (2025)“, Kap. 2; Kap. 3.
[46] EU, AI Act (2024)**,** Art. 12; Art. 13; Art. 14.
[47] Datenschutzkonferenz (DSK), „DSK, OH KI & Datenschutz (2024)“**,** S. 5–7.
[48] Eigene Erhebung; vgl. Kap. 4.2.2–4.2.3.

5.1.2 Datenschutz, Barrierefreiheit und digitale Souveränität

Aus der DSGVO folgen Zweckbindung, Datenminimierung und Integrität der Verarbeitung. Für Behörden heißt das: steuerbare Datenweitergabe an externe KI-Dienste (Filter/Pseudonymisierung, minimal notwendige Inhalte, bevorzugt EU-Infrastruktur) und fein granulare Rechte.[49] [50] Parallel gelten strenge Anforderungen an Barrierefreiheit nach der Web-Zugänglichkeitsrichtlinie und der BITV 2.0, zum Beispiel Alt-Texte, eine saubere semantische Struktur und zugängliche PDFs. Diese Aspekte haben Vorrang bei KI-gestützten Fällen, bleiben aber in der redaktionellen Verantwortung.[51]

Die Befragten konkretisieren diese Vorgaben durch Forderungen nach EU-Standorten, Datenminimierung, Pseudonymisierung und transparenter Kontextweitergabe. Für die Konzeption bedeutet dies, dass jede externe Verarbeitung sichtbar autorisiert, protokolliert und auf den aufgabenspezifisch minimalen Kontext begrenzt wird.[52]

Zur Gewährleistung von Offenheit und Interoperabilität nutzt die Architektur das MCP als offenes Protokoll mit deklarativen Tool-Verträgen, während auf CMS-Seite TYPO3-Workspaces und die Versionierung als Governance-Anker dienen.

5.1.3 Use Cases für MCP mit TYPO3 im öffentlichen Sektor

A) Mehrsprachigkeit und Leichte Sprache

MCP-Tools lesen Inhalte strukturiert aus, erzeugen Übersetzungsentwürfe und legen neue Sprachvarianten nur im Entwurfs-Workspaces an welche dann von der Redaktion geprüft und ggf. veröffentlicht werden. Erfolgreich ist der Ansatz, wenn die vorgesehenen Sprachen abgedeckt sind, die KI-Unterstützung klar gekennzeichnet wird und die Durchlaufzeiten sinken.[53]

---

[49] Bundesministerium des Innern, „BMI, Leitlinien KI Bundesverwaltung (2025)“, Kap. 3.
[50] Datenschutzkonferenz (DSK), „DSK, OH KI & Datenschutz (2024)“, S. 3–5; 10–12.
[51] Deutschland, „Barrierefreie-Informationstechnik-Verordnung (BITV 2.0)“, 12. September 2011, Abschn. §3, https://www.gesetze-im-internet.de/bitv_2_0/ letzter Zugriff 14.09.2011 13:36 Uhr.
[52] Eigene Erhebung; vgl. Kap. 4.2.2.
[53] TYPO3 contributors, „TYPO3 Workspaces — Workspaces Main Documentation“, 22. August 2025, https://docs.typo3.org/c/typo3/cms-workspaces/main/en-us letzter Zugriff 14.09.2025 14:36 Uhr.

B) Barrierefreiheit umfasst Alt-Texte, sprachliche Vereinfachung und PDF-Prüfungen. Ein MCP-Tool erkennt fehlende Alternativtexte und schlägt passende Formulierungen vor. Als Praxisbeleg dient die TYPO3 Extension ai_filemetadata. In gleicher Weise lassen sich sprachliche Vereinfachungen und PDF-Prüfungen automatisieren, stets mit einer abschließenden menschlichen Kontrolle. Akzeptanzkriterien sind eine breite Abdeckung der Medientypen, ein klarer Review-Pfad je Vorschlag sowie nachweislich höhere Erfüllungsquoten gemäß BITV und BFSG.[54] [55]

C) Konsistenz und Aktualität
Die Lösung erkennt veraltete Bezeichnungen, Terminologieinkonsistenzen und fehlende Metadaten. Korrekturen werden nicht unmittelbar veröffentlicht, sondern zunächst als Entwürfe angelegt. Die Redaktion prüft die Vorschläge, gibt sie frei und veröffentlicht sie anschließend. Die Governance erfolgt über Workspaces und die Versionierung stellt Nachvollziehbarkeit sicher und ermöglicht das Rückgängigmachen von Änderungen. Akzeptanzkriterien sind geschlossene Metadatenlücken, eine konsistente Terminologie über die definierten Inhalte hinweg und ein dokumentierter Reviewpfad.[56] [57]

D) Die Architektur erreicht RAG-Fähigkeit und Multi-Channel, indem MCP kontrolliert Kontext aus TYPO3 bereitstellt und für Retrieval eine Headless-JSON-Schnittstelle via EXT:headless nutzt.[58] Der Ansatz gilt als erfolgreich, wenn die Tool-Verträge eindeutig festgelegt sind, das Monitoring von Kontexttreffern und Antwortqualität kontinuierlich läuft und Entwurfs- und Live-Stände klar getrennt bleiben. [59]

---

[54] Marketing Factory Digital GmbH, *marketing-factory/ai-filemetadata*, PHP, 30. August 2024; Marketing Factory Digital GmbH, released 12. September 2025, https://github.com/marketing-factory/ai-filemetadata letzter Zugriff 14.09.2025 14:55 Uhr.
[55] Deutschland, „BITV 2.0“, Abschn. §3.
[56] TYPO3 contributors, „TYPO3 Workspaces“, Kap. „Workspaces“.
[57] TYPO3 contributors, „TYPO3 Workspaces“, Kap. „Versioning“.
[58] Tymoteusz Motylewski u. a., „EXT:Headless — Headless Main Documentation“, TYPO3 Documentation, 17. April 2025, https://docs.typo3.org/p/friendsoftypo3/headless/main/en-us letzter Zugriff 14.09.2025 18:18.
[59] Anthropic, „MCP Spec v0.2.0“, Abschn. „Tools“.

E) Im Redaktionsalltag unterstützt die Assistenz mit Vorschlägen für Meta-Beschreibungen, strukturierte Daten, Schlagwörter und Titelvarianten. Jede Aktion wird protokolliert[60] und lässt sich vollständig zurücknehmen.[61] Akzeptanzkriterien sind kürzere Erstellzeiten, vollständig ausgefüllte Felder sowie bessere Auffindbarkeit und Konsistenz.

Als Praxisreferenz jenseits von TYPO3 zeigt das Drupal-Ökosystem mit einem MCP-Server-Modul die Realisierbarkeit CMS-seitiger MCP-Funktionen und damit die Übertragbarkeit der Use Cases.[62]
In allen Fällen ist entscheidend, dass Entwürfe erst nach menschlicher Prüfung veröffentlicht werden. Darüber hinaus braucht es eine klare und begründete Weitergabe von Kontext, eine lückenlose Protokollierung und spürbare Fortschritte bei Geschwindigkeit und Qualität. Diese Kriterien wurden in der Befragung wiederholt benannt und bilden die Grundlage für die spätere Evaluation.[63]

5.1.4 Referenzflüsse der Konzeption

Die Konzeption unterscheidet zwei Pfade. Der Leseweg in KI-Kontexte stellt veröffentlichte Inhalte kontrolliert bereit. Der Entwurfsweg in das CMS unterstützt die Redaktion mit Vorschlägen, die als Entwürfe entstehen und erst nach Prüfung veröffentlicht werden. Nachfolgend soll mit jeweils drei Beispielen für Lese- und Entwurfsweg ein besseres Verständnis erzeugt werden.
Beispiele für den Leseweg:

1. Dieser Anwendungsfall beschreibt ein Bürger*innen FAQ zu Verwaltungsleistungen. Ein KI-Client ruft über eine MCP Resource[64]

[60] TYPO3 contributors, „The Logging Framework (Developer Guide)“, TYPO3 Documentation, 12. September 2025, https://docs.typo3.org/m/typo3/reference-coreapi/main/en-us/ApiOverview/Logging letzter Zugriff 14.09.2025 19:29 Uhr.
[61] TYPO3 contributors, „TYPO3 Workspaces“, „Versioning“.
[62] Drupal contributors, „Drupal - Model Context Protocol“, Drupal.Org, 26. November 2024, https://www.drupal.org/project/mcp letzter Zugriff 18.09.2025 22:14 Uhr.
[63] Eigene Erhebung; vgl. Kap. 4.2.1–4.2.3.
[64] Anthropic, „MCP Spec v0.2.0“, Resources.

ausschließlich die für die Antwort benötigten Felder bereits veröffentlichter Leistungsbeschreibungen ab. Zum Beispiel Titel, Kurztext, Haupttext und die offizielle URL. Optional nutzt der Client die Headless[65] Ausgabe als Korpus für Retrieval[66], um belegte Passagen in die Antwort aufzunehmen. Akzeptanzkriterien sind präzise Quellenangaben, der konsequente Ausschluss von Entwurfsinhalten im Kontext und eine vorab definierte maximale Antwortzeit.

2. Dieser Anwendungsfall beschreibt den Umgang mit Presseanfragen zu Verordnungen. Ein KI-Client bezieht über eine MCP Resource kontextminimiert ausschließlich Metadaten bereits veröffentlichter Verordnungen, zum Beispiel Titel, Gültigkeitszeitraum, Veröffentlichungsdatum und die offizielle URL. Bei umfangreichen Dokumenten nutzt der Client zusätzlich Retrieval mit Passagenzitaten. Akzeptanzkriterien sind korrekte Datumsangaben, präzise Pinpoint Zitate und keinerlei Zugriffe auf gesperrte oder nicht veröffentlichte Bereiche.[67] [68]
3. Dieser Anwendungsfall behandelt Anfragen zu Öffnungszeiten und Kontaktinformationen. Ein Chatbot bezieht über eine MCP Resource kontextminimiert ausschließlich die für die Antwort benötigten Felder bereits veröffentlichter Standortseiten. Zum Beispiel Öffnungszeiten, Anschrift, Telefonnummer, E-Mail und die offizielle URL. Zugriffe erfolgen nur auf den Live Bestand, Entwurfsinhalte bleiben ausgeschlossen. Ein Monitoring erfasst Trefferquote und nachträgliche Korrekturen zur Qualitätssteuerung[69]. Akzeptanzkriterien sind ein hoher Anteil korrekter Antworten und transparente, verständliche Fehlermeldungen.

Beispiele für den Entwurfsweg:

1. Dieser Anwendungsfall beschreibt die Generierung von Alternativtexten für Bilder. Ein Tool erstellt für ein Asset einen Alt-Textvorschlag als Entwurf im zulässigen Workspace[70]. Die Redaktion prüft den Vorschlag, verbessert ihn bei Bedarf und veröffentlicht ihn über den regulären Freigabeprozess.

[65] Motylewski u. a., „EXT:Headless“.
[66] Lewis u. a., „Retrieval-Augmented Generation“.
[67] Anthropic, „MCP Spec v0.2.0“, Resources.
[68] Lewis u. a., „Retrieval-Augmented Generation“.
[69] TYPO3 contributors, „TYPO3 Logging“.
[70] TYPO3 contributors, „TYPO3 Workspaces“, Versioning.

Versionierung und Protokollierung erfassen Akteur[71], Zeitpunkt sowie alten und neuen Wert. Akzeptanzkriterien sind messbar bessere Barrierefreiheitswerte[72] und lückenlose Audit-Einträge.

2. Dieser Anwendungsfall beschreibt Vorschläge für Meta Beschreibung und Titelvarianten einer Seite. Ein Tool erzeugt beide als nicht destruktive Entwürfe[73] im zulässigen Workspace und ergänzt eine Evidenzliste, die die verwendeten Quellen aufführt. Die Redaktion prüft die Entwürfe, übernimmt oder verwirft sie und veröffentlicht erst nach der redaktionellen Freigabe[74]. Akzeptanzkriterien sind verkürzte Erstellzeiten, konsistent gepflegte Felder und die jederzeit mögliche Rücknahme.
3. Dieser Anwendungsfall beschreibt die Erstellung einer Fassung in Einfacher Sprache für Bürgertexte. Ein Tool erstellt auf Grundlage der veröffentlichten Inhalte eine vereinfachte Sprachvariante als nicht destruktiven Entwurf im vorgesehenen Workspace. Der Live-Bestand bleibt dabei unverändert. Mindestens zwei prüfende Personen bewerten Verständlichkeit und fachliche Korrektheit, dokumentieren ihre Prüfschritte und geben erst nach erfolgreicher Prüfung zur Veröffentlichung frei. Akzeptanzkriterien sind verbesserte Lesbarkeitskennzahlen, vollständig dokumentierte Prüfvorgänge und eine für Nutzer klar erkennbare Kennzeichnung der Assistenz.

5.1.5 Risiken und Gegenmaßnahmen

Um Halluzinationen und andere Qualitätsmängel zu vermeiden, wird ein klarer Human in the Loop umgesetzt. Inhalte werden erst nach fachlicher Prüfung veröffentlicht, Schreibrechte gelten zunächst nur im Entwurfsmodus und sowohl das eingesetzte Modell als auch die verwendeten Prompts werden vollständig protokolliert, damit Ergebnisse überprüfbar bleiben.[75] [76] Beim Datenschutz gelten Datenminimierung und das konsequente Ausklammern vertraulicher Teilbäume

[71] TYPO3 contributors, „TYPO3 Logging“.
[72] World Wide Web Consortium (W3C), „Web Content Accessibility Guidelines (WCAG) 2.2“, W3C Technical Reports, 12. Dezember 2024, https://www.w3.org/TR/WCAG22/ letzter Zugriff 19.09.2025 22:17 Uhr.
[73] Anthropic, „MCP Spec v0.2.0“, Tools.
[74] TYPO3 contributors, „TYPO3 Workspaces“.
[75] Bundesministerium des Innern, „BMI, Leitlinien KI Bundesverwaltung (2025)“, Kap. 2; Kap. 3.
[76] TYPO3, *Logging*, Kap. Writers**/Audit**.

als Grundprinzipien. Wo es nötig ist, werden Daten pseudonymisiert. Die Verarbeitung findet in EU-Hostingumgebungen statt, und die Dokumentation orientiert sich an den Vorgaben der DSK.[77] Um Vendor Lock-in zu vermeiden, baut die Architektur auf das Model Context Protocol als offenen und anbieterneutralen Standard. Integrationen laufen dabei über klar definierte Tool-Verträge und nicht über proprietäre Kopplungen.[78]
„Langfristige Nutzbarkeit durch offene Protokolle wie MCP statt proprietärer APIs […] Die Austauschbarkeit der LLMs muss gewährleistet sein, um zwischen externen oder lokalen Instanzen zu wechseln, ohne dass das System neu entwickelt werden muss."
Frank Nägler.[79]
Barrierefreiheitsdefizite werden adressiert, indem eine KI-Assistenz als verpflichtender Schritt im Upload- und Publikationsprozess für Medien und PDFs etabliert wird und Kennzahlen die Erfüllungsgrade messbar machen.[80]

5.1.6 Abgeleitete Anforderungen an die Systemkonzeption

1. Human-in-the-Loop ist verbindlich. KI schreibt ausschließlich in Workspaces und Inhalte gehen erst nach einer menschlichen Freigabe live. Mehrstufige Prüfungen sind möglich.[81]
2. Rechte folgen dem Minimalprinzip. Ein technischer KI-Benutzer erhält nur eng begrenzte Lese- und Schreibrechte und sensible Bereiche bleiben getrennt.[82]
3. Jede Aktion ist nachvollziehbar. Das TYPO3 Logging Framework protokolliert Vorgänge eindeutig und die Protokolle bleiben persistent erhalten.[83]

[77] Datenschutzkonferenz (DSK), „DSK, OH KI & Datenschutz (2024)", S. 10–12; 3–5.
[78] Anthropic, „MCP Spec v0.2.0".
[79] Eigene Erhebung (Expertenumfrage), Frank Nägler, Antwort auf Frage 13. Gibt es weitere Punkte, die Sie in Bezug auf MCP oder KI in CMS-Kontexten für wichtig halten?
[80] Deutschland, „BITV 2.0", Abschn. § 3.
[81] TYPO3 contributors, „TYPO3 Workspaces", Abschn. Versioning and Workspaces.
[82] TYPO3 contributors, „TYPO3 Workspaces", Abschn. Versioning and Workspaces.
[83] TYPO3, *Logging*, Kap. Writers/Audit.

4. Datenschutz ist von Anfang an mitgedacht. Es gelten Zweckbindung und Datenminimierung, Filter schützen vertrauliche Inhalte und die Verarbeitung erfolgt bevorzugt in EU-Umgebungen.[84] [85]
5. Schnittstellen sind offen und versioniert. MCP-basierte Tool-Verträge sichern Modell- und Anbieterneutralität.[86]
6. Das System ist RAG-bereit. Neben MCP steht eine Headless-JSON-Ausgabe zur Verfügung, Kontext-Tools sind definiert und Kontexttreffer sowie Antwortqualität werden laufend überwacht.[87]

Diese sechs Anforderungen sind durch die Befragung abgesichert, vor allem Human-in-the-Loop, minimale Rechte, transparente Kontextweitergabe, Auditierbarkeit und die Austauschbarkeit über Tool-Verträge. Sie fassen die in 4.2 verdichteten Erwartungen zu einer prüfbaren Sollarchitektur zusammen.[88]

[84] Bundesministerium des Innern, „BMI, Leitlinien KI Bundesverwaltung (2025)“, Kap. 3.
[85] Datenschutzkonferenz (DSK), „DSK, OH KI & Datenschutz (2024)“, S. 10–12; 3–5.
[86] Anthropic, „MCP Spec v0.2.0“.
[87] Motylewski u. a., „EXT:Headless“.
[88] Eigene Erhebung; vgl. Kap. 4.2–4.3.

5.2 Konzeptentwicklung zur MCP-Integration in TYPO3

5.2.1 Architekturmodell der MCP-Integration

Unter einem Architekturmodell wird hier eine abstrahierte, konsistente Darstellung der Bauteile, Schnittstellen, Datenflüsse und Kontrollpunkte verstanden, die die MCP-Integration in TYPO3 fachlich und technisch erklärt. Das Modell bleibt technologieoffen und versionierbar und dient als Referenzrahmen für Implementierungen. Es benennt Komponenten und Beziehungen, begründet die Governance-Anker Human-in-the-Loop, minimale Rechte, Versionierung und Auditierbarkeit und zeigt, wie diese in Lese- und Schreibpfaden wirksam werden. Abbildung 1 zeigt das Architekturmodell der MCP-Integration in TYPO3. Es benennt Komponenten, ihre Beziehungen und die Kontrollpunkte (Lesen: nur Live, Schreiben: nur Entwurf im Workspace, Token/Berechtigungen).

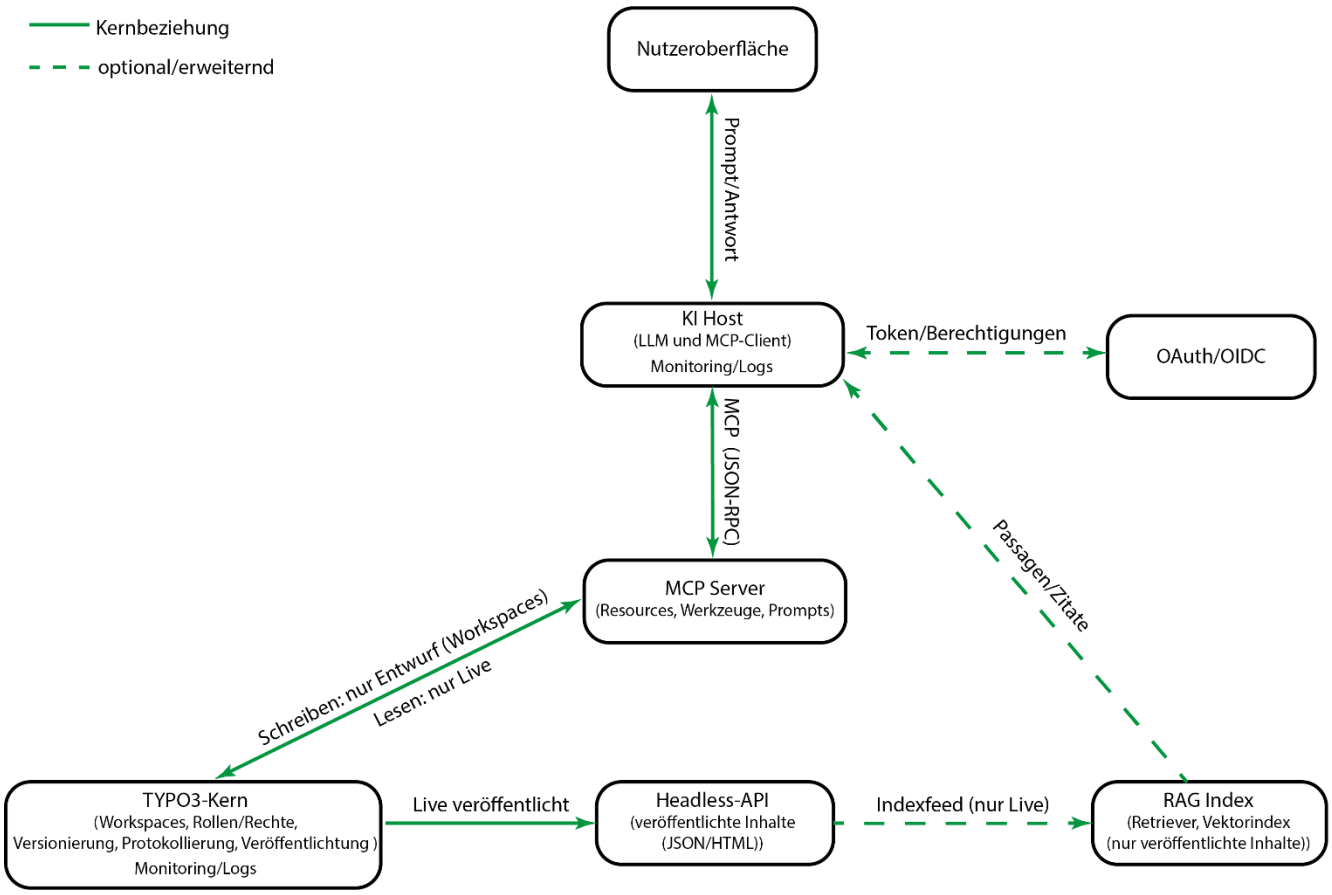


Abbildung 1: Architekturmodell mit Kontrollpunkten: Live-only Read, Draft-only Write (Workspace), Token/Scopes per OAuth/OIDC (optional), RAG als Erweiterung der Live-Headless-Ausgabe, Monitoring/Logs betriebsbegleitend.

Die Integration des Model Context Protocol (MCP) in TYPO3 verfolgt das Ziel, KI-Assistenz standardisiert, kontrolliert und auditierbar in die redaktionellen Abläufe einzubetten. Aus den in Kapitel 5.1 abgeleiteten Anforderungen, also Transparenz, menschliche Aufsicht, Datenschutz und Datenminimierung, digitale Souveränität sowie Anschlussfähigkeit an Retrieval-Verfahren, ergibt sich ein Systemkonzept, das offene Schnittstellen über MCP mit den Governance-Funktionen von TYPO3 verknüpft. So können KI-gestützte Inhalte kontextbewusst erzeugt, überprüft und veröffentlicht werden.[89]
Auf diese Weise werden die in der Befragung priorisierten Kontrollmechanismen systematisch abgebildet, nämlich Schreiben ausschließlich im Entwurfsmodus, menschliche Freigaben, kontextminimiertes Lesen und durchgängiges Logging.[90]
Zentraler Baustein ist MCP als offener Protokollstandard, der eine einheitliche Kopplung zwischen einem KI-Client (z. B. LLM-Assistent) und TYPO3 als MCP-Server ermöglicht. MCP legt Rollen und das JSON-RPC-Nachrichtenformat fest und definiert die Zusammenhänge von Resources (kontextuelle Daten), Tools (ausführbare Funktionen) und Prompts (vordefinierte Interaktionsmuster). So werden Kontextzugriffe und Aktionen anbieter- und modellneutral standardisiert; fragmentierende Einzellösungen werden vermieden und Austauschbarkeit gefördert, ohne den CMS-Kern zu verändern.[91] MCP verankert zugleich grundlegende Sicherheits- und Kontrollprinzipien. Dazu gehören User Consent and Control, Data Privacy, Tool Safety und LLM Sampling Controls. Sie verlangen explizite Autorisierungen, klare Sichtbarkeitsgrenzen und Freigaben über die Benutzeroberfläche, bevor Daten offengelegt oder Tools gestartet werden.[92]
Damit die KI nur unterstützend wirkt und nicht eigenständig veröffentlicht, greift das Konzept konsequent auf die Workspaces von TYPO3 zurück. Alle Schreibvorgänge der KI erscheinen zunächst als Entwurf, während Live-Inhalte unverändert bleiben, bis die Redaktion sie freigibt. Workspaces stellen revisionssichere Entwürfe, Vorschau sowie mehrstufige Review-/Publish-Prozesse bereit und erzwingen so Human-in-the-Loop als harte Schranke zwischen Entwurf und Veröffentlichung.[93]

[89] Anthropic, „MCP Spec v0.2.0“.
[90] Eigene Erhebung; vgl. Kap. 4.2.2–4.2.3.
[91] Anthropic, „MCP Spec v0.2.0“, Architecture.
[92] Anthropic, „MCP Spec v0.2.0“.
[93] TYPO3 contributors, „TYPO3 Workspaces“.

Damit grenzt sich das Systemkonzept von schnellen KI-Plugins ab, die direkt in Live Felder schreiben. Die MCP Integration arbeitet innerhalb des Redaktionsprozesses und respektiert das bestehende Berechtigungs- und Freigabesystem.

Nachvollziehbarkeit entsteht durch eine strukturierte Protokollierung im TYPO3 Logging Framework. Jeder KI-Schritt mit Angaben zu Akteur oder Service Konto, Tool, Ziel Datensatz, Workspace, Ergebnis und Korrelation ID wird in Audit Trails festgehalten, die geprüft und exportiert werden können. Writer wie Datei, Syslog oder externe Systeme und ergänzende Produktionsleitfäden sichern eine verlässliche Umsetzung im Betrieb.[94]

Die Protokollierung ergänzt die Sicherheitsprinzipien des MCP und stellt die Auditierbarkeit der Prozesse sicher, was für öffentliche Stellen essenziell ist.[95]

Ein zweiter Pfeiler ist die digitale Souveränität. TYPO3 integriert keine KI direkt in den Core, sondern stellt klare Schnittstellen bereit, ganz nach dem Prinzip „Interfaces instead of Integration". So bleiben Modelle und Provider (On-Prem, EU-Cloud, Open-Source) austauschbar, ohne die CMS-Logik neu aufzubauen. Das schützt vor Vendor Lock-in und ist eine Voraussetzung für staatliche Handlungsfähigkeit und Datenhoheit.[96] Auf strategischer Ebene betont der IT-Planungsrat, dass die öffentliche Verwaltung Wahlfreiheit, Kontrolle über Daten und Unabhängigkeit von Einzelanbietern sichern muss; Abhängigkeiten gefährden die Steuerungsfähigkeit, weshalb offene Standards und substituierbare Komponenten zentrale Leitplanken sind.[97]

Schließlich ist das Systemkonzept für RAG ausgelegt. Wissensgestützte Antworten entstehen zuverlässiger, wenn ein LLM gezielt autorisierte und aktuelle Inhalte abrufen kann. MCP eröffnet kontrollierte Lesepfade aus TYPO3 über sogenannte Resources. Studien zeigen, dass RAG Modelle durch eine zusätzlich eingebundene nicht parametrische Wissensbasis faktentreuer und spezialisierter antworten als rein parametrische Ansätze.[98]

---

[94] TYPO3 contributors, „TYPO3 Logging".
[95] Anthropic, „MCP Spec v0.2.0", Abschn. Security and Trust&Safety.
[96] Nägler, „TYPO3 V14 – AI Integrations".
[97] IT-Planungsrat, „Föderale Digitalstrategie für die Verwaltung – Teil 1: Zukunftsbild und Leitlinien; Teil 2: Zielbilder der Schwerpunktthemen", IT-Planungsrat, 26. März 2025, Abschn. 4.1.4, https://www.it-planungsrat.de/fileadmin/it-planungsrat/der-it-planungsrat/foederale_digitalstrategie/250513_IT_PLR_Foerderale_Digitalstrategie_Zukunftsbild_Leitlinien.pdf letzter Zugriff 19.09.2025 13:51 Uhr.
[98] Lewis u. a., „Retrieval-Augmented Generation".

Für öffentliche Websites bedeutet dies, dass KI-Antworten auf geprüften und freigegebenen Inhalten basieren. Entwürfe sind dabei deutlich vom Live Bestand abgegrenzt. Im vorgeschlagenen Systemkonzept werden offene, standardisierte Schnittstellen (MCP) mit den Governance-Funktionen von TYPO3 verknüpft, sodass KI-Assistenz standardisiert eingebettet, revisionssicher nachvollzogen und menschlich gesteuert werden kann. Die in Abschnitt 5.1 beschriebenen Anforderungen sind erfüllt. Dazu zählen die kontrollierte Weitergabe von Kontext mit expliziter Autorisierung durch MCP, ein Entwurfsmodus mit nachfolgender Freigabe in Workspaces, eine vollständige Protokollierung über das Logging Framework und die flexible Einbindung von KI Diensten über klar definierte Interfaces als Ausdruck digitaler Souveränität.[99] [100] [101] [102]

5.2.2 Schnittstellen und Datenfluss

Nachfolgend soll kurz die Schrittfolge für einen generischen Ablauf dargestellt werden. Das Beispiel ist nicht use-case-spezifisch, sondern bildet den Standardverlauf einer Interaktion bestehend aus KI Client, MCP Server und TYPO3 von der Auslösung bis zur Veröffentlichung und zum Audit ab. Aus Kapitel 5.1.5 werden die entsprechenden Anforderungen als Kurzform in Klammern dahinter notiert.

1. Anstoß durch Redaktion (z. B. „übersetze Seite X"): MCP-Client wird gezielt gestartet. (Anforderungen: 1, 3)
2. Authentisierung & Rollenprüfung im CMS; technischer KI-User mit Minimalrechten. (Anforderung: 2)
3. Explizite Freigabe/Consent für den konkreten Vorgang (Scope). (Anforderungen: 4, 3)
4. Resource-Listing (MCP): nur freigegebene Objekte/Felder erscheinen. (Anforderungen: 2, 4)
5. Minimaler Read (MCP Resource Read): nur benötigte Felder (Titel, Inhalt, ausgewählte Metadaten). (Anforderung: 4)

[99] Anthropic, „MCP Spec v0.2.0".
[100] TYPO3 contributors, „TYPO3 Workspaces".
[101] TYPO3 contributors, „TYPO3 Logging".
[102] IT-Planungsrat, „Föderale Digitalstrategie 2025".

6. Optionaler Wissens-Read für RAG: veröffentlichte Inhalte zusätzlich headless als JSON bereitstellen; Entwürfe bleiben nur via MCP-Resource zugänglich. (Anforderungen: 6, 4)
7. Tool-Ausführung (MCP Tool Call) mit strikt begrenztem Kontext; Prompt/Parameter werden versioniert protokolliert. (Anforderungen: 3, 5)
8. Draft-Write: Ergebnis wird ausschließlich in den Workspace geschrieben und nicht direkt Live auf die entsprechende Seite gestellt. (Anforderung: 1)
9. Vorschau & Begründung: KI-Vorschlag inkl. Herkunft/Kontext sichtbar; ggf. Iteration mit engen Scopes. (Anforderungen: 3, 4)
10. Review (Human-in-the-Loop): redaktionelle Prüfung/Überarbeitung im Entwurf. (Anforderung: 1)
11. Freigabe & Publish über bestehenden TYPO3-Workflow; Live erst nach Approval. (Anforderungen: 1, 3)
12. Audit & Monitoring: vollständiges Logging (Akteur/Service-Konto, Resource, Tool, Korrelation-ID) + Qualitätstracking (Kontexttreffer/Antwortqualität für RAG). (Anforderungen: 3, 6)

Der Abschnitt beschreibt den Interaktionsablauf zwischen KI-Client und TYPO3 vom Anstoß einer redaktionellen Aktion über den kontextminimierten Lesezugriff bis zum Schreiben im Entwurfsmodus, zur anschließenden Prüfung und Veröffentlichung sowie zur Protokollierung. Zugleich werden die in 5.2.1 abgeleiteten Kontrollpunkte im operativen Ablauf verankert: Autorisierung, Datenminimierung, Schreiben ausschließlich im Entwurfsmodus, Auditierbarkeit und Human-in-the-Loop. Diese Abfolge entspricht den in der Befragung formulierten Erwartungen an Entwurfswege, Kontexttransparenz und Nachvollziehbarkeit. Ausgangspunkt ist immer eine redaktionelle Aktion wie etwa das Übersetzen einer Seite oder das Ergänzen von Alt Texten. Die KI arbeitet dabei nur mit Entwürfen, alle Schreibvorgänge landen im Workspace und Live Inhalte bleiben bis zur Freigabe unverändert. Auf diese Weise sind Änderungen reversibel, nachvollziehbar und organisatorisch abgesichert.[103]
Damit die Einbettung sowohl datenschutzgerecht als auch prozesssicher gelingt, orientiert sich der Kontextaustausch streng am Minimalprinzip. Die KI erhält nur aufgabenspezifische Ausschnitte, die ausdrücklich freigegeben sind. Jeder Zugriff

[103] TYPO3 contributors, „TYPO3 Workspaces“.

auf Daten und jedes Ausführen von Tools erfordert eine sichtbare, bewusste Autorisierung, und wenn Kontexte erweitert oder erneut verwendet werden, gilt dies als neue Freigabe.[104] Dies adressiert die geforderte Datenminimierung und Verantwortlichkeitszuordnung aus der Befragung.
In der Umsetzung werden die Tool- und Resource-Verträge so zugeschnitten, dass der Client nur die benötigten Felder/Objekte anfordern kann (Scopes), idealerweise zusätzlich beschränkt auf Entwurfsstände.[105]
Im Leseschritt fordert der Client die minimale Datenmenge an, die er für die Bearbeitung benötigt (z. B. Seitentitel, Fließtextfelder, ausgewählte Metadaten). Welche Objekte und Felder gelesen werden dürfen, ist über Resource Definitionen und Scopes klar begrenzt. Sensible Bereiche und nicht benötigte Felder bleiben verborgen.[106] Für wissensintensive Aufgaben kann der Flow zusätzlich veröffentlichte Inhalte headless als JSON bereitstellen. Entwürfe sind hingegen ausschließlich über freigegebene Ressourcen zugänglich. So bleibt die Grenze zwischen Live-Wissen und Redaktionswissen gewahrt.[107] Empirisch erhöht Retrieval-Augmented Generation (RAG) die Faktentreue, weil nicht-parametrisches Wissen kontrolliert eingebunden wird.[108]
Im Schreibschritt ruft der Client ein Schreib-Tool auf, das nur in den Workspace schreibt (Entwurf).[109] Das Ergebnis sind neue oder veränderte Datensätze ohne direkte Auswirkung auf den Live Bestand. Versionierung und Rücknahme bleiben erhalten, und auch die bestehenden Rechte und Workflows gelten weiter.[110] Die Architektur gewährleistet Human-in-the-Loop, da eine Liveschaltung erst nach Review und Publish erfolgt. Bis dahin bleiben KI Vorschläge sichtbar, können überarbeitet oder verworfen werden und lassen sich jederzeit rückgängig machen.[111]
Parallel entstehen Audit-Trails: Jeder Schritt (Freigabe, Lesezugriff, Tool-Ausführung, Draft-Schreiben, Review) wird strukturiert protokolliert (Akteur/Service-Konto, Zielobjekt, Workspace, Ergebnis, Korrelation-ID). Das Logging Framework von TYPO3 stellt PSR-3 konforme Writer und Exporte bereit,

[104] Anthropic, „MCP Spec v0.2.0“.
[105] Anthropic, „MCP Spec v0.2.0“, Abschn. Listing Resources, Reading Resources.
[106] Anthropic, „MCP Spec v0.2.0“, Authorization, 2.7 Error Handling.
[107] Motylewski u. a., „EXT:Headless“.
[108] Lewis u. a., „Retrieval-Augmented Generation“.
[109] TYPO3 contributors, „TYPO3 Workspaces“, Abschn. Administration/Workspace.
[110] TYPO3 contributors, „TYPO3 Workspaces“, Abschn. Administration/Versioning.
[111] TYPO3 contributors, „TYPO3 Workspaces“, Abschn. Administration/Creating a custom workspace.

sodass Nachweise geprüft und exportiert werden können. Betriebsrichtlinien wie Rotation, Retention und Forwarding halten die Balance zwischen Prüfbarkeit und Datenminimierung.[112] Security-by-Design flankiert den Flow auf Infrastrukturebene: Härtung der Integrationspunkte, Scope-Trennung, Monitoring der Protokolle sowie geübte Incident-Response-Prozesse (EU-weit empfohlener Mehrschichten-Rahmen).[113] Nationale Hinweise konkretisieren die sichere Nutzung (Kontextsparsamkeit, qualifizierte Bewertung, organisatorische Vorkehrungen) und lassen sich unmittelbar in Freigabe- und Redaktionsabläufe übersetzen.[114]

Der Flow ist anbieter- und modellneutral gestaltet. TYPO3 bietet Schnittstellen anstelle einer fest eingebauten KI nach dem Prinzip „Interfaces instead of Integration". Dadurch bleibt die Logik des Datenflusses stabil, auch wenn Modelle oder Dienste wechseln, sei es On Prem, in einer EU-Cloud oder mit Open Source. Dieser Grad an Entkopplung stärkt die digitale Souveränität und verhindert Lock in.[115] Auf der Governance-Ebene ist menschliche Aufsicht nicht nur wünschenswert, sondern im Sinne europäischer Vorgaben als Best Practice so auszugestalten, dass sie wirksam eingreifen kann, etwa durch Freigabeschritte, Rollentrennung sowie Versionierung mit Rücknahmeoptionen.[116]

Ergebnis: Der Schnittstellen- und Datenfluss verknüpft Consent & Minimalprinzip (vor/während des Lesens), Draft-Only-Schreiben (bei der Tool-Ausführung), Human-in-the-Loop (im Review/Publish), Auditierbarkeit (durchgängig) und Sicherheitskontrollen (quer über alle Schritte).

Für wissensintensive Aufgaben speist der Flow RAG-Pipelines kontrolliert mit autorisiertem Kontext, ohne die Sicherheitsgrenzen des CMS zu unterlaufen. Die Abfolge bildet die operative Brücke zwischen der konzeptionellen Zielsetzung (5.2.1) und den technischen Ausführungen der folgenden Unterkapitel. Für wissensintensive Aufgaben wird damit zugleich die in der Befragung

---

[112] TYPO3 contributors, „TYPO3 Logging".

[113] European Union Agency for Cybersecurity (ENISA), „Multilayer Framework for Good Cybersecurity Practices for AI | ENISA", Multilayer Framework for Good Cybersecurity Practices for AI, 21. Februar 2024, https://www.enisa.europa.eu/publications/multilayer-framework-for-good-cybersecurity-practices-for-ai letzter Zugriff 19.09.2025 19:57 Uhr.

[114] Bundesamt für Sicherheit in der Informationstechnik (BSI), „Künstliche Intelligenz sicher nutzen", Bundesamt für Sicherheit in der Informationstechnik, 18. Juni 2024, https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/Broschueren/Wegweiser_Checklisten_Flyer/Brosch_A6_Kuenstliche_Intelligenz.html?nn=132646 letzter Zugriff 19.09.2025 20:28 Uhr.

[115] Nägler, „TYPO3 V14 – AI Integrations".

[116] Europäische Union, „EU, AI Act (2024)".

hervorgehobene Voraussetzung einer kuratierten, semantisch ausgezeichneten Wissensbasis erfüllt.[117]

5.2.3 Umsetzungsszenario

Im Umsetzungsszenario wird TYPO3 um einen MCP-Server ergänzt, der klar definierte Resources (für kontextminimierte Lesezugriffe) und Tools (für strikt begrenzte Schreibvorgänge) bereitstellt. Der KI-Client entdeckt diese Verträge, ruft sie mit Parametern auf und erhält nur jene Inhalte, die für die jeweilige Aufgabe freigegeben sind.[118] [119] Im CMS erfolgen Änderungen ausschließlich als Entwürfe in Workspaces und werden erst nach einer redaktionellen Prüfung veröffentlicht. Das Livesystem bleibt während der Bearbeitung unverändert.[120] [121]
Die Rollen- und Rechtevergabe erfolgt nach dem Minimalprinzip: ein technisches Service-Konto besitzt nur die für den Vorgang erforderlichen Zugriffsrechte (Module, Seitenbäume, Felder). So wird sichergestellt, dass Resource-Reads und Tool-Writes stets innerhalb der autorisierten Scopes erfolgen.[122] Optional kann die Organisation mit Rollenmustern arbeiten, um Berechtigungen sauber zu kapseln.[123]
Die Interaktionsebene nutzt ein schlankes RPC-Vorbild: Der Client übermittelt einen JSON-RPC-Request (Methode, Parameter, Korrelation über id), der MCP-Server verarbeitet und antwortet transportagnostisch.[124] Schreibende Aufrufe sind als Tools ausgestaltet und werden versioniert protokolliert (Prompt, Parameter, Ergebnis). Parallel führt TYPO3 ein PSR-3-kompatibles Logging mit

[117] Eigene Erhebung; vgl. Kap. 4.2.1–4.2.3.
[118] Anthropic, „MCP Spec v0.2.0“, Abschn. Resources.
[119] Anthropic, „MCP Spec v0.2.0“, Abschn. Tools.
[120] TYPO3 contributors, „TYPO3 Workspaces“.
[121] TYPO3 contributors, „TYPO3 Workspaces“, Administration/Workspaces/Publishing and swapping.
[122] TYPO3 contributors, „Permissions Management — TYPO3 Explained Main Documentation“, 19. September 2025, https://docs.typo3.org/m/typo3/reference-coreapi/main/en-us/Administration/PermissionsManagement/ letzter Zugriff 19.09.2025 20:39 Uhr.
[123] TYPO3 contributors, „Roles — TYPO3 Explained Main Documentation“, 19. September 2025, https://docs.typo3.org/m/typo3/reference-coreapi/main/en-us/ApiOverview/Backend/AccessControl/Roles/ letzter Zugriff 19.09.2025 20:48 Uhr.
[124] JSON-RPC Working Group, „JSON-RPC 2.0“.

Korrelation-IDs, sodass alle Schritte – Freigabe, Lesezugriff, Tool-Call, Draft-Write, Review – nachvollziehbar bleiben.[125] [126]
Für wissensintensive Aufgaben wie faktennahe Übersetzungen oder die Erstellung von FAQs kann der Client zusätzlich veröffentlichte Inhalte über eine JSON basierte Headless API beziehen. Entwürfe sind dagegen nur über MCP Resources erreichbar. Die Erweiterung EXT:headless stellt dafür eine JSON-API aus regulären TYPO3-Inhalten bereit, konfigurierbar über TypoScript. Auf diese Weise lässt sich Retrieval Augmented Generation kontrolliert mit autorisiertem und veröffentlichtem Kontext versorgen. Studien belegen, dass RAG die Faktentreue im Vergleich zu rein parametrischen Modellen erhöht.[127]
Zugriffe des Service Kontos auf geschützte HTTP Ressourcen werden durch etablierte Autorisierungsflüsse abgesichert. In typischen Setups wird dafür OAuth 2.0 genutzt, dass einem Drittclient einen zeitlich begrenzten und zweckgebundenen Zugriff erlaubt.[128] In Verbindung mit den oben beschriebenen MCP-Verträgen und TYPO3-Workspaces entsteht so ein durchgängig kontrollierbarer Lese-/Schreibpfad mit Human-in-the-Loop, Auditierbarkeit und klarer Trennung von Entwurf und Live.

5.3 Vergleich mit bestehenden Lösungen

Aufbauend auf dem in Abschnitt 5.2.2 beschriebenen Referenzablauf und den in 5.1.5 formulierten Anforderungen wird das hier entwickelte TYPO3-Szenario den MCP-basierten Integrationen in Drupal und WordPress gegenübergestellt. Maßgeblich sind dabei die Fragen, wie Zugriffskontrolle und Scopes, Entwurf-vor-Live, Rollen/Rechte, Audit/Logging, Tool-/Prompt-Handhabung sowie eine optionale Headless-Anbindung für wissensintensive Aufgaben konzeptionell umgesetzt sind. Das TYPO3-Modell verzahnt MCP-Resources (für minimalen Lesezugriff) und MCP-Tools (für strikt begrenzte Schreibvorgänge) eng mit den nativ verfügbaren Workspaces: Schreiboperationen landen ausschließlich im Entwurfsmodus, werden versioniert sichtbar und gelangen erst nach

[125] TYPO3 contributors, „TYPO3 Logging“.
[126] PHP-FIG, „PSR-3: Logger Interface - PHP-FIG“, PHP-FIG, zugegriffen 19. September 2025, https://www.php-fig.org/psr/psr-3/ letzter Zugriff 19.09.2025 23:57 Uhr.
[127] Motylewski u. a., „EXT:Headless“.
[128] Hardt, *OAuth 2.0 (RFC 6749)*.

redaktioneller Freigabe in den Live-Bestand.[129] Die Drupal-Integration ergänzt das Standard-Rollenmodell um eine explizite MCP-Berechtigung und setzt freizugebende Content-Types auf Opt-in. Nur bewusst exponierte Typen erscheinen als MCP-Resources, und die Authentisierung kann auf ein definiertes Benutzerkonto gelegt werden.[130] Die Entwurfsdisziplin ist dabei kein Modulbestandteil. Änderungen verhalten sich wie reguläre Entity-Operationen und benötigen, falls gewünscht, die Content-Moderation von Drupal, die außerhalb des MCP-Modulumfangs liegt.[131]

Die WordPress-Integration stützt sich auf die Capability-Vererbung des angemeldeten Benutzers und unterstützt STDIO wie HTTP. Üblich sind JWT oder Anwendungskennwörter. Das Plugin erlaubt feingranulare CRUD-Gates, wodurch sich Schreibpfade als Entwurf konfigurieren lassen. Erzwungen wird der Entwurfsstatus allerdings nicht durch die MCP-Schicht selbst.[132] Ergänzend stellt die Implementierung Audit-/Debug-Funktionen bereit, sodass sicherheitsrelevante Vorgänge und Token-Lebenszyklen nachvollziehbar sind.[133]

Im direkten Vergleich zeigt sich: Datenminimierung und Scoping sind in allen drei Ansätzen möglich, aber unterschiedlich „hart". Drupal legt mit Permission und Opt in klare Eintrittstore für MCP fest, überlässt die Entwurfs- und Freigabeprozesse aber dem Workflow des CMS. WordPress bietet eine feine Steuerung über Capabilities bis hin zu CRUD Schaltern, doch ob Entwürfe konsequent genutzt werden, bleibt eine Frage der jeweiligen Policy.[134] [135] Das hier entworfene TYPO3-Modell verankert Draft-Only dagegen systemisch. Workspaces sorgen dafür, dass KI-Schreiben standardmäßig nicht live-wirksam ist und stets den redaktionellen Prüf- und Freigabeweg passiert.[136]

---

[129] TYPO3 contributors, „TYPO3 Workspaces", Administration/Workspaces/Publishing and swapping.

[130] Drupal contributors, „What Is MCP?", Drupal MCP, Setup & Configure, zugegriffen 20. September 2025, https://drupalmcp.io/en/introduction/what-is-mcp/ letzter Zugriff 20.09.2025 00:06 Uhr.

[131] Drupal contributors, „Content Moderation – Overview", Drupal.Org, 8. April 2025, https://www.drupal.org/docs/8/core/modules/content-moderation/overview letzter Zugriff 20.09.2025 11:03 Uhr.

[132] Automattic, *WordPress MCP — README*, PHP, 29. April 2025; Automattic, released 19. September 2025, https://github.com/Automattic/wordpress-mcp letzter Zugriff 20.09.2025 11:15.

[133] Automattic, *MCP WordPress Remote — README*, TypeScript, 29. April 2025; Automattic, released 18. September 2025, https://github.com/Automattic/mcp-wordpress-remote letzter Zugriff 20.09.2025 11:30 Uhr.

[134] Drupal contributors, „Drupal - What Is MCP?", Setup & Configure.

[135] Automattic, *WordPress MCP*.

[136] TYPO3 contributors, „TYPO3 Workspaces", Administration/Workspaces/Publishing and swapping.

Für Audit und Monitoring stehen in TYPO3 PSR-3-konforme Logging-Bausteine bereit (Logger, Writer, Processor) einschließlich Produktionsrichtlinien. KI-bezogene Lese-, Schreib- und Freigabeereignisse lassen sich damit revisionssicher dokumentieren.[137] [138] [139]

Auch bei der Tool/Prompt-Handhabung setzen die Systeme unterschiedliche Akzente: WordPress bietet mit dem MCP Adapter eine formale Registratur für Tools, Resources und Prompts. Deklaration und Rechtekopplung erfolgen standardisiert.[140] Drupal erweitert den Server über MCP-Plugins und Submodule (zum Beispiel mcp_extra, mcp_dev_tools). Das TYPO3-Szenario bindet Tool-Verträge eng an Workspaces und Rechte und sieht Prompt-Versionierung prozessual vor, also über Protokolle und Artefakte, statt über eine zentrale Prompt-Registry.[141]

5.4 Potenzial und Herausforderungen

Vor dem Hintergrund der in Kapitel 5.1 bis 5.3 beschriebenen Architektur erscheint MCP in TYPO3 geeignet, Potenziale an den definierten Kontrollpunkten zu adressieren. Erstens kann Potenzial durch zentrale Pflege und Wiederverwendung von Inhalten über Sites und Sprachen entstehen. TYPO3 erlaubt Entwürfe, Freigaben und Versionierung in Workspaces, sodass Redaktionen konsistent publizieren können. Das kann Qualität und Nachvollziehbarkeit in Behörden begünstigen.[142]

Zweitens könnten standardisierte Tool-Verträge und ein prüfender Entwurfsmodus die Zeit bis zur Veröffentlichung reduzieren, weil Assistenzfunktionen repetitive Arbeit wie Metadatenvorschläge oder Zusammenfassungen vorbereiten und die Redaktion systematisch prüft. MCP stellt dafür ein einheitliches Protokoll bereit und entkoppelt KI-Tools von der CMS-Logik.[143]

---

[137] TYPO3 contributors, „TYPO3 Logging“.
[138] PHP-FIG, „PSR-3“.
[139] TYPO3 contributors, „Log Writers — TYPO3 Explained Main Documentation“, 19. September 2025, https://docs.typo3.org/m/typo3/reference-coreapi/main/en-us/ApiOverview/Logging/Writers/ letzter Zugriff 20.09.2025 11:54 Uhr.
[140] Pascal Birchler, „MCP Adapter“, *WordPress AI*, 17. Juli 2025, https://make.wordpress.org/ai/2025/07/17/mcp-adapter/ letzter Zugriff 20.09.2025 11:48 Uhr.
[141] Drupal contributors, „Drupal - What Is MCP?“, Setup & Configure.
[142] TYPO3 contributors, „TYPO3 Workspaces“.
[143] Anthropic, „MCP Spec v0.2.0“.

Drittens kann die Trennung von fachlicher Struktur im CMS und austauschbaren MCP-Tools zur Zukunftsfähigkeit beitragen. Neue Modelle lassen sich anbinden, ohne die redaktionellen Abläufe grundsätzlich neu zu entwerfen.[144]

Diese Effekte treten nur ein, wenn die betrieblichen Rahmenbedingungen passen. Zu den Hauptherausforderungen zählen die initiale Konfiguration im Zusammenspiel mit Workspaces, Rechten und mehrsprachigen Seitenbäumen sowie die Verankerung des Entwurfsprinzips im Redaktionsalltag. Ohne verbindliche Rollen und Prüfschritte steigt das Risiko ungeprüfter Änderungen.[145]

Technisch ist eine klare Trennung von Lesezugriffen mit Datenminimierung und Schreiboperationen im Entwurf nötig. Lücken bei Schnittstellenverträgen und Protokollierung erzeugen blinde Flecken in der Revision. Das TYPO3-Logging sollte daher gezielt konfiguriert und ausgewertet werden.[146]

Der öffentliche Sektor bringt zusätzliche Anforderungen. Datenschutz und Governance verlangen eindeutige Zwecke, strenge Protokollierung und klare Freigabeketten. Bei MCP-gestützten Workflows sind Risiken durch missbräuchliche Eingaben und unkontrollierte externe Kontexte zu berücksichtigen. Leitlinien empfehlen in solchen Fällen die Minimierung von Berechtigungen, klare Prüfschritte sowie robuste Betriebsmodelle, um Manipulationen vorzubeugen. [147] [148]

Implikationen für die Umsetzung: Pilotierung in abgegrenzten Inhaltsbereichen mit messbaren Kriterien für Zeitgewinn, Konsistenz und Fehlerquote. Werkzeuge nur dort aktivieren, wo Datenminimierung gewährleistet ist. Veröffentlichung ausschließlich über Entwurf und Review führen. Tool-Verträge dokumentieren und in Schulungen überführen. So lassen sich Potenziale realisieren, während Risiken durch Governance, Protokollierung und schrittweises Ausrollen beherrschbar bleiben.

[144] Motylewski u. a., „EXT:Headless“.
[145] TYPO3 contributors, „Backend User Groups — TYPO3 Explained Main Documentation“, 19. September 2025, https://docs.typo3.org/m/typo3/reference-coreapi/main/en-us/Administration/UserManagement/Groups/Index.html#backend-user-groups letzter Zugriff 20.09.2025 10:57 Uhr.
[146] TYPO3 contributors, „TYPO3 Logging“.
[147] Bundesministerium des Innern, „BMI, Leitlinien KI Bundesverwaltung (2025)“.
[148] Datenschutzkonferenz (DSK), „DSK, OH KI & Datenschutz (2024)“.

5.5 MCP Ablaufbeispiel

Ziel dieses Abschnitts ist es, einen vollständigen Ablauf von der Nutzeranfrage bis zur Veröffentlichung darzustellen. Er setzt das in Kap. 5.2 entwickelte Architekturmodell operativ um. Der MCP-Client wird als Komponente auf der KI-Seite verstanden; seine Platzierung im Host oder als benachbarte Orchestrierung ist für das Protokoll unerheblich. Maßgeblich sind die in Kap. 5.2.1 beschriebenen Bausteine und Kontrollpunkte.

Das Systemkonzept stützt sich auf mehrere zentrale Komponenten und Annahmen. Die Nutzeroberfläche bildet den Einstiegspunkt für redaktionelle Aktionen. Ein KI Host mit LLM und MCP Client kommuniziert mit dem MCP Server, der über definierte Resources, Tools und optional Prompts verfügt. Der TYPO3 Kern stellt Workspaces für Entwürfe bereit und gewährleistet Rollen und Rechte, Versionierung, Logging sowie den Publish Workflow. Für veröffentlichte Inhalte steht zusätzlich eine Headless API zur Verfügung. Retrieval Augmented Generation greift über einen Retriever und einen Vektorindex auf einen Korpus veröffentlichter Inhalte zu. Identität und Berechtigungen werden durch OAuth oder OIDC abgesichert. Ergänzend sorgen Monitoring und Telemetrie für Transparenz im Betrieb. Das Service Konto arbeitet mit minimalen Scopes, während Policies am MCP Server Draft only, Quoten und Pflichtfelder erzwingen. Der RAG Index enthält ausschließlich veröffentlichte Inhalte. Einzelne Details sind in den Kapiteln 5.2.1 und 5.2.2 vertieft beschrieben.

Lesesequenz (Schritte 1–10): Die Schritte folgen dem Architekturmodell und den Ausführungen in Kap. 5.2.2, Schritte (4)–(7).
(1) Der/Die Nutzer*in stellt die Aufgabe in der Oberfläche, der Prompt gelangt an den KI-Host.
(2) Der Host bewertet, ob zusätzlicher Kontext erforderlich ist, und nutzt dafür die MCP-Client-Komponente.
(3) Der Client bezieht oder erneuert ein OAuth bzw. OIDC Token mit passenden Scopes.
(4) Der Client ruft am MCP Server eine Resource auf, um autoritative Inhalte aus TYPO3 in kontextminimierter Form zu lesen. Übertragen werden nur die benötigten Felder wie Titel, strukturierte Auszüge und Metadaten.

(5) Der Server prüft Token, Policy und berechtigte Felder und liest die Daten aus dem TYPO3-Kern.
(6) Die Resource liefert den minimalen Kontext an den Client zurück.
(7) Optional kann der Client auch veröffentlichte Inhalte über die Headless API in HTML oder JSON abrufen. Entwürfe sind über diese Schnittstelle nicht sichtbar.
(8) Der RAG Index wird regelmäßig und unabhängig vom Prompt aus der Headless API aktualisiert. Wenn nötig, fragt der Client den Retriever ab und bekommt Passagen mit Quellenangaben zurück.
(9) Der Client baut aus Resource-Ergebnissen, optionalen Headless-Antworten und RAG-Passagen einen Prompt mit Belegen und übergibt ihn an die LLM.
(10) Die LLM generiert einen Vorschlag, zum Beispiel eine Zusammenfassung, Alt-Texte oder strukturierte Metadaten, und gibt ihn an den Client zurück.
Zwischenergebnis. Ein Vorschlag liegt vor, ist noch nicht im CMS gespeichert, Quellen sind belegbar, es wurden nur minimal erforderliche Daten gelesen, Entwürfe waren zu keinem Zeitpunkt über die Headless-API sichtbar. Vgl. Kap. 5.2.2.

Schreibsequenz (Schritte 11–16): Die Schritte folgen Kap. 5.2.2, Schritte (7)–(11).
(11) Der Client prüft Pflichtfelder und Evidenzlisten gegen das vereinbarte Tool-Schema.
(12) Der Client ruft am MCP-Server ein Tool auf, um den Vorschlag als Entwurf in TYPO3 zu speichern.
(13) Der Server prüft Token, Scopes, Quoten und Pflichtfelder und schreibt den Datensatz in den Workspace Entwurf. Versionierung und Logging erfassen Vorgang, Zeitpunkt, Quelle und Akteur.
(14) Die Redaktion sichtet den Entwurf im TYPO3 Kern und überarbeitet ihn bei Bedarf. Jede Änderung erzeugt einen neuen Entwurfsstand in der Versionierung.
(15) Nach fachlicher und formaler Prüfung erteilt die Redaktion die Freigabe im Publish Workflow.
(16) Mit der Veröffentlichung wird der Inhalt über die Headless API sichtbar. Der RAG Crawler übernimmt die freigegebene Version in den Index.
Ergebnis: Die Veröffentlichung ist eine menschliche Entscheidung im TYPO3-Workflow, die KI publiziert nicht selbst, alle Schritte sind nachvollziehbar, versioniert und protokolliert. Vgl. Kap. 5.2.2.

Governance greift entlang beider Sequenzen. Identität und Rechte vergibt der Client über OAuth oder OIDC, der MCP Server prüft bei jedem Aufruf Token und Scopes. Policies am MCP Server erzwingen Draft only, Quoten und Pflichtfelder. Im TYPO3 Kern werden Rollen und Rechte durchgesetzt, Versionierung und Audit Log erfassen Schreibvorgänge, Reviews und Veröffentlichungen. Korrelations IDs verknüpfen Client Aufrufe und Protokolle. Monitoring und Telemetrie erfassen Durchlaufzeiten, Fehlerraten, Quoten und Auslastung. Vgl. Kap. 5.2.1 und Kap. 5.2.2.

Die Schnittstellen und Datenformate sind klar definiert. Resource und Tool Aufrufe erfolgen über JSON RPC mit eindeutigen Methodennamen und überprüfbaren Schemas. Die Headless API stellt veröffentlichte Inhalte als JSON oder HTML bereit und kann optional JSON LD Strukturen liefern. Der RAG Retriever greift auf einen Vektorindex aus der Headless API zu und gibt Passagen mit Quellenangaben zurück. Vgl. Kap. 2, Kap. 5.2.1 und Kap. 5.2.2.

Fehlende Pflichtfelder beim Tool Aufruf führen zu einer Ablehnung mit Hinweis. Der Client ergänzt die Angaben und startet den Aufruf erneut. Ist der Scope unzureichend, wird der Zugriff verweigert und Rollenprofil oder Token werden angepasst oder der Vorgang beschränkt sich auf eine Leseoperation. Veraltete Kontexte im RAG Index werden durch zusätzliche Resource Reads abgesichert und als möglicherweise überholt markiert. Konflikte im Workspace löst die Redaktion auf, wobei neue Entwurfsstände entstehen. Vgl. Kap. 5.2.2.

6. Diskussion und Ausblick

6.1 Zusammenfassung und Beantwortung der Forschungsfragen

Die Arbeit beschreibt ein Architekturmodell, das MCP in TYPO3 so einbindet, dass KI an bestehende Redaktionsabläufe anschließt, statt sie zu umgehen. Beim Lesen gilt Datenminimierung, der Client erhält nur den Kontext, der wirklich gebraucht wird. Beim Schreiben entsteht zuerst ein Entwurf im Workspace, die Veröffentlichung bleibt eine redaktionelle Entscheidung. Durchgängiges Logging macht Schritte und Zuständigkeiten nachvollziehbar. Aus Analyse und Befragung ergibt sich ein vorsichtig positives Bild. Tempo und Qualität können steigen, wenn Rollen und Rechte sauber eingerichtet sind, Versionierung genutzt wird und Protokolle verlässlich ausgewertet werden.
Zur ersten Forschungsfrage zeigt das Modell, dass sich KI in TYPO3 beherrschbar anbinden lässt, solange Entwürfe statt Direktveröffentlichung gelten, Freigaben verbindlich sind, Kontext nur gezielt weitergegeben wird und alle Schritte protokolliert werden. Die zweite Frage richtet den Blick auf Kontrollen. Notwendig sind ein klarer Human in the Loop, transparente Kontextweitergabe und lückenlose Nachweise. Das senkt das Risiko von Fehlpublikationen und erhöht die Prüfbarkeit im Alltag. Die dritte Frage betrifft die Voraussetzungen im System. Erforderlich sind Entwurfsarbeit mit nachgelagerter Freigabe, fein abgestufte Rechte, konsequente Versionierung und verlässliches Logging. Hinzu kommen offene Schnittstellen und gepflegte Tool Verträge sowie Anforderungen aus dem öffentlichen Umfeld wie Datenschutz, Transparenz, Barrierefreiheit und Nachvollziehbarkeit. Die befragten Personen halten das im TYPO3 Kontext für umsetzbar, wenn Governance, Logging und Schulungen fest verankert sind.

6.2 Methodische Grenzen und Validität

Die Arbeit ist bewusst als Konzept angelegt. Das Architekturmodell stützt sich auf Fachliteratur, eine Befragung von Personen mit Erfahrung und eine technische Betrachtung der beteiligten Systeme. Die Ergebnisse wirken in sich stimmig, sie wurden aber noch nicht über längere Zeit in echten Redaktionen erprobt.

Aussagen zur Wirksamkeit sind deshalb gut begründete Erwartungen und keine Effekte aus einer Feldstudie.
Die Empirie beruht auf einer kleinen Gruppe von Befragten, die dem Thema Content Management und dem hier betrachteten Technologieumfeld sehr nah sind. Das bringt fachliche Tiefe, kann aber den Blick in Richtung Expertensicht verschieben. Hinzu kommt, dass sich eher Menschen mit starkem Interesse an KI und Automatisierung beteiligen. Wie gut sich die Ergebnisse übertragen lassen, hängt daher von den konkreten Bedingungen vor Ort ab, zum Beispiel von bestehenden Redaktionsprozessen, vom Ausbildungsstand der Mitarbeitenden und vom Reifegrad des Systems.
Untersucht wird vor allem mit qualitativen Einschätzungen, strukturierten Anforderungen und abgeleiteten Prinzipien. Solche Aussagen zu Effizienz, Qualität und Akzeptanz sind wertvoll, ersetzen aber keine Messwerte. Für eine belastbare Prüfung braucht es klare Kennzahlen wie Durchlaufzeiten, Korrekturraten oder die Güte von Metadaten und Texten. Diese Größen sollten vor der Einführung und nach der Einführung erhoben werden, damit die Ergebnisse belastbar sind.
Auch die interne Validität spielt eine Rolle. Verbesserungen können ebenso aus mehr Prozessdisziplin, aus Schulungen oder aus parallelen Infrastrukturprojekten stammen. Das Modell setzt außerdem ein sauberes Rechtekonzept, klare Freigaben und vollständiges Logging voraus. Fehlen diese Bausteine, lassen sich beobachtete Effekte nicht eindeutig dieser Architektur zuschreiben.
Die Architektur ist am Beispiel TYPO3 entwickelt und am öffentlichen Sektor ausgerichtet. Die Leitprinzipien sind grundsätzlich übertragbar, wenn ein Entwurfsmechanismus für Schreibvorgänge vorhanden ist, Versionierung verlässlich arbeitet, Rollen und Rechte fein abgestuft sind und Protokolle lückenlos geführt werden. Systeme ohne diese Grundlagen können das Konzept nur eingeschränkt nutzen. Mehrsprachige Sites, Mandantenstrukturen und vorhandene Freigabeprozesse beeinflussen die Übertragbarkeit zusätzlich.
Die Rolle der Forschenden kann zu Bestätigungsneigungen führen, wenn eigene Annahmen oder Präferenzen das Urteil beeinflussen.
Zur Stärkung der Validität verbindet die Arbeit Befragung, Fachliteratur und Systemanalyse. Begriffe und Erfolgsmaße sind definiert und werden im Hauptteil konsistent verwendet. Für den nächsten Schritt bietet sich ein Pilotprojekt mit Vorher-Nachher-Vergleich an. Messgrößen, Zeitpunkte und

Vergleichsbedingungen sollten vorab festgelegt werden. Eine zweite Codierung qualitativer Aussagen und eine Prüfung der Protokolle auf Vollständigkeit erhöhen die Zuverlässigkeit. Im Ergebnis ist das Modell fachlich schlüssig und anschlussfähig. Seine Gültigkeit im Betrieb hängt jedoch von klaren Governance Regeln, von messbaren Erfolgskriterien und von einer kontrollierten Einführung ab. Erst ein Pilot mit definierten Kennzahlen kann die erwarteten Effekte bestätigen oder widerlegen.

6.3 Implikationen für Praxis und weitere Forschung

Für die Praxis zeichnet sich ein klarer Einstieg ab. Redaktionen arbeiten mit KI grundsätzlich im Entwurf und geben erst nach Prüfung frei. So ist es in Kapitel 5.2.1 beschrieben. Die Weitergabe von Kontext bleibt sichtbar und begründet, damit Prüfschritte nachvollziehbar sind. Darauf verweisen die Leitprinzipien in Kapitel 5.2. Hilfreich ist ein kurzer Leitfaden mit Rollen, Zuständigkeiten und Beispielen. Er reduziert Unsicherheit im Alltag und knüpft an die Bedarfe aus Kapitel 4 an. Schulungen behandeln nicht nur die Bedienung, sondern auch typische Fehlerbilder, Eskalationswege und Kriterien für die Annahme oder Ablehnung von KI-Vorschlägen. Die Kontrollpunkte aus Kapitel 5.2 dienen dabei als Orientierung. Ein kleines Glossar mit zentralen Begriffen stärkt die gemeinsame Sprache und beugt Missverständnissen vor. Das passt zur Terminologie in Kapitel 5.1.

Technisch lohnt sich ein zentrales Management der Tool Verträge. Versionen, Freigabeabläufe und eine kurze Dokumentation erleichtern Pflege und Betrieb. Siehe dazu das Vertragskonzept in Kapitel 5.2.1. Lesezugriffe folgen konsequent der Datenminimierung. Schreibzugriffe landen ausschließlich als Entwurf im Workspace und werden protokolliert. Der Ablauf in Kapitel 5.2.1 beschreibt das im Detail. Quoten, Limits und Zeitfenster steuern die Nutzung. Secrets werden sicher verwaltet. Ein Monitoring hält Ausfälle, Antwortzeiten, Kontextgrößen und Fehlmuster fest. Diese Systemaspekte sind in Kapitel 5.2 angelegt. Für sensible Inhalte empfiehlt sich ein erhöhter Prüfmodus mit strengeren Freigaben und zusätzlichen Protokolleinträgen. Das folgt den Governance Punkten aus Kapitel 5.2. Staging mit Rollback schützt vor Fehlinhalten und unterstützt kontrollierte Releases, wie in Kapitel 5.2.1 beschrieben. Mehrsprachigkeit und

Mandantenfähigkeit gehören früh in die Planung, weil sie Rechte, Workflows und die Auswahl zulässiger Quellen beeinflussen. Das leitet sich aus den Anforderungen in Kapitel 5.1 ab.
Governance und Compliance bilden den Rahmen. Zuständigkeiten zwischen Redaktion, IT und Datenschutz werden schriftlich festgelegt und mit dem Rechtekonzept aus Kapitel 5.2 verzahnt. Ein kurzer Prüfplan vor dem Go Live klärt, welche Inhalte mit KI bearbeitet werden dürfen, wie lange Protokolle aufbewahrt werden und wer auf welche Metriken zugreifen darf. Das knüpft an Logging und Metriken aus Kapitel 5.2 an. Für behördliche Redaktionen sind Barrierefreiheit, Nachvollziehbarkeit und eine revisionssichere Dokumentation verpflichtend. In der Beschaffung und in Verträgen sind offene Schnittstellen, klare Servicelevel und realistische Exit Möglichkeiten wichtig, damit spätere Anpassungen nicht blockiert werden. Bei Entwicklung und Betrieb sind die Vorgaben der EU KI-Verordnung zu beachten.
Für die Einführung zählen messbare Kriterien. Geeignet sind die Durchlaufzeit von der Anlage bis zur Freigabe, die Zahl der Korrekturrunden, der Anteil übernommener KI Vorschläge, die Qualität von Metadaten, die Fehlerquote in Publikationen und der Aufwand für Nachbesserungen. Diese Größen lehnen sich an die Erfolgsmaße aus Kapitel 5.1 an. Sie werden vor der Einführung erhoben und im Pilot fortlaufend gemessen. Ein kleines Dashboard schafft Transparenz. Ein monatliches Review entscheidet über Anpassungen an Verträgen, Workflows und Schulungen. Diese Steuerungslogik ist in Kapitel 5.2 angelegt. Erfolgreiche Teile des Piloten werden schrittweise ausgeweitet. Für die begleitende Forschung eignet sich ein Design mit Messung vor und nach der Einführung, anschlussfähig an die Ausgangslage aus Kapitel 4. Ergänzend helfen kontrollierte Vergleiche zwischen klassischer Arbeit ohne KI und dem hier beschriebenen Ansatz, wie in Kapitel 5.2.1 skizziert.
Ein zweites Thema ist die Ausgestaltung der Tool Verträge. Varianten mit unterschiedlichen Kontextbudgets, Prüflogiken und Antwortformaten lassen sich im Pilot vergleichen. Messgrößen aus Kapitel 5.1 und die Steuerungslogik aus Kapitel 5.2 geben den Rahmen vor. Replikationen in anderen CMS prüfen die Übertragbarkeit, wie in Kapitel 5.3 angelegt. Qualitative Verfahren wie Leitfadeninterviews und Tagebücher ergänzen die Messwerte und zeigen, wie Redaktionen im Alltag entscheiden. Langfristig ist ein Katalog wiederverwendbarer Vertragsbausteine denkbar. Er verknüpft Bausteine mit

dokumentierten Effekten und Risiken und folgt dem Kataloggedanken aus Kapitel 5.2.

6.4 Ausblick

Die nächsten Schritte bauen direkt auf den in Kapitel 4 identifizierten Bedarfen und den in Kapitel 5 entwickelten Leitprinzipien auf. Kurzfristig empfiehlt sich ein Pilot in einer klar abgegrenzten Redaktionseinheit. Der Pilot verwendet das in Kapitel 5.2.1 beschriebene Ablaufmodell mit Entwurfsarbeit, nachgelagerter Freigabe, sichtbarer Kontextweitergabe und durchgängigem Logging. Vor dem Start werden Messgrößen festgelegt, die in Kapitel 5.1 bereits skizziert sind, zum Beispiel Durchlaufzeiten, Zahl der Korrekturrunden, Qualität von Metadaten und Anteil übernommener Vorschläge. Ein kleines Dashboard schafft Transparenz über Nutzung, Fehlerbilder und Ausfälle. Parallel entstehen ein kompakter Redaktionsleitfaden und kurze Schulungen, die typische Entscheidungsregeln und Eskalationswege vermitteln. Wo sensible Inhalte betroffen sind, wird ein erhöhter Prüfmodus konfiguriert, der strengere Freigaben und zusätzliche Protokollierung nutzt, vgl. Kapitel 5.2.

Mittelfristig wird der Funktionsumfang entlang, der in Kapitel 4 genannten Aufgaben erweitert. Zunächst werden Tool Verträge versioniert und im Sinne eines Katalogs gepflegt, damit wiederkehrende Aufgaben mit geprüften Parametern bearbeitet werden können. Die technische Basis aus Kapitel 5.2 wird um Monitoring, Quoten und ein belastbares Secret Management ergänzt. Mehrsprachigkeit, Mandantenfähigkeit und Site Handling werden früh in die Governance aufgenommen, weil sie Rollen, Rechte und zulässige Quellen beeinflussen. Für inhaltliche Qualität werden Feedback Schleifen etabliert, in denen Redaktionen Annahmen, Ablehnungen und Nachbesserungen begründen. Diese Hinweise fließen in die Weiterentwicklung von Verträgen und in Schulungen zurück. Zur Absicherung der Ergebnisse werden Vorher Nachher Vergleiche mit den in Kapitel 5.1 definierten Kennzahlen fortgeführt.

Langfristig zielt die Architektur auf einen stabilen Betrieb und auf Übertragbarkeit.

> „Einen großen Nutzen hätte es sicherlich um Daten zwischen Kommunen/Ländern/Bund einheitlicher und

einfacher abzurufen und auszutauschen. Dadurch könnten bürokratische Hürden abgebaut und die Verwaltung entlastet werden."[149]

AutoDudes (Manuel Schnabel & André Kraus). Wenn der Pilot seine Ziele erreicht, lässt sich der Ansatz Schritt für Schritt auf weitere Bereiche einer Behörde ausweiten. Das kann eine Abteilung sein, ein Referat oder später ganze Häuser in der öffentlichen Verwaltung. Für andere CMS bleibt das Grundmuster nutzbar, solange Workflows, Versionierung, Rechte und Protokolle vorhanden sind, wie in Kapitel 5.3 beschrieben. Für die Forschung lohnt ein Katalog mit wiederverwendbaren Vertragsbausteinen, jeweils mit dokumentierten Effekten, Risiken und Grenzen. Sinnvoll ist außerdem die gezielte Anbindung fachlicher Wissensquellen, die Kontexte sparsam bereitstellen und den Lesezugriff aus Kapitel 5.2 respektieren. Abschließend sollte die Governance regelmäßig auf den Prüfstand. Rollen, Freigaben, Protokolle und Messgrößen wachsen so mit der Praxis mit und neue Anforderungen lassen sich geordnet aufnehmen.

[149] Eigene Erhebung (Expertenumfrage), **AutoDudes (Manuel Schnabel & André Kraus)**, Antwort auf Frage 8

Literaturverzeichnis

Anthropic. „Introducing the Model Context Protocol". Anthropic News, 25. November 2024. https://www.anthropic.com/news/model-context-protocol.

Anthropic. „Model Context Protocol Specification". Model Context Protocol, 18. Juni 2025. https://modelcontextprotocol.io/specification/2025-06-18.

Anthropic. „What Is the Model Context Protocol (MCP)?" Model Context Protocol. Zugegriffen 12. September 2025. https://modelcontextprotocol.io/docs/getting-started/intro.

Automattic. *MCP WordPress Remote — README*. TypeScript. 29. April 2025; Automattic, released 18. September 2025. https://github.com/Automattic/mcp-wordpress-remote.

Automattic. *WordPress MCP — README*. PHP. 29. April 2025; Automattic, released 19. September 2025. https://github.com/Automattic/wordpress-mcp.

Birchler, Pascal. „MCP Adapter". *WordPress AI*, 17. Juli 2025. https://make.wordpress.org/ai/2025/07/17/mcp-adapter/.

Bitkom e. V. „KI-Nutzung boomt – aber die Angst vor Abhängigkeit vom Ausland ist groß". Bitkom, 5. Mai 2025. https://www.bitkom.org/Presse/Presseinformation/KI-Nutzung-boomt-Angst-vor-Abhaengigkeit-Ausland-gross.

Bray, Tim. *The JavaScript Object Notation (JSON) Data Interchange Format*. RFC 8259. RFC Editor, Internet Standard (STD 90) Dezember 2017. https://doi.org/10.17487/RFC8259.

Brown, Tom B., Benjamin Mann, Nick Ryder, u. a. „Language Models are Few-Shot Learners". arXiv:2005.14165. Preprint, arXiv, 22. Juli 2020. https://doi.org/10.48550/arXiv.2005.14165.

Bundesamt für Sicherheit in der Informationstechnik (BSI). „Künstliche Intelligenz sicher nutzen". Bundesamt für Sicherheit in der Informationstechnik, 18. Juni 2024. https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/Broschueren/Wegweiser_Checklisten_Flyer/Brosch_A6_Kuenstliche_Intelligenz.html?nn=132646.

Bundesministerium des Innern. „Leitlinien für den Einsatz Künstlicher Intelligenz in der Bundesverwaltung". Bundesministerium des Innern, 27. März 2025.

https://www.bmi.bund.de/SharedDocs/downloads/DE/publikationen/themen/moderne-verwaltung/ki/BMI25020-leitlinien-ki-bundesverwaltung.pdf.

Chelli, Mikaël, Jules Descamps, Vincent Lavoué, u. a. „Hallucination Rates and Reference Accuracy of ChatGPT and Bard for Systematic Reviews: Comparative Analysis". *Journal of Medical Internet Research* 26 (Mai 2024): e53164. https://doi.org/10.2196/53164.

Datenschutzkonferenz (DSK). „Orientierungshilfe ‚KI und Datenschutz', Version 1.0". 6. Mai 2024. https://www.datenschutzkonferenz-online.de/media/oh/20240506_DSK_Orientierungshilfe_KI_und_Datenschutz.pdf.

Deutschland. „Barrierefreie-Informationstechnik-Verordnung (BITV 2.0)". 12. September 2011. https://www.gesetze-im-internet.de/bitv_2_0/.

DMK E-BUSINESS GmbH. *typo3-mkcontentai*. PHP. 17. Mai 2023; DMK E-BUSINESS GmbH, released 28. Juli 2025. https://github.com/DMKEBUSINESSGMBH/typo3-mkcontentai.

Drupal contributors. „Content Moderation – Overview". Drupal.Org, 8. April 2025. https://www.drupal.org/docs/8/core/modules/content-moderation/overview.

Drupal contributors. „Drupal - Model Context Protocol". Drupal.Org, 26. November 2024. https://www.drupal.org/project/mcp.

Drupal contributors. „What Is MCP?" Drupal MCP. Zugegriffen 20. September 2025. https://drupalmcp.io/en/introduction/what-is-mcp/.

Europäische Union. „Verordnung (EU) 2024/1689". 12. Juli 2024. https://eur-lex.europa.eu/eli/reg/2024/1689/oj/eng.

European Union Agency for Cybersecurity (ENISA). „Multilayer Framework for Good Cybersecurity Practices for AI | ENISA". Multilayer Framework for Good Cybersecurity Practices for AI, 21. Februar 2024. https://www.enisa.europa.eu/publications/multilayer-framework-for-good-cybersecurity-practices-for-ai.

Guu, Kelvin, Kenton Lee, Zora Tung, Panupong Pasupat, und Ming-Wei Chang. „REALM: Retrieval-Augmented Language Model Pre-Training". arXiv:2002.08909. Preprint, arXiv, 10. Februar 2020. https://doi.org/10.48550/arXiv.2002.08909.

Hardt, Dick. *The OAuth 2.0 Authorization Framework*. Request for Comments RFC 6749. Internet Engineering Task Force, 2012. https://doi.org/10.17487/RFC6749.

Hou, Xinyi, Yanjie Zhao, Shenao Wang, und Haoyu Wang. „Model Context Protocol (MCP): Landscape, Security Threats, and Future Research Directions". arXiv:2503.23278. Preprint, arXiv, 6. April 2025. https://doi.org/10.48550/arXiv.2503.23278.

IT-Planungsrat. „Föderale Digitalstrategie für die Verwaltung – Teil 1: Zukunftsbild und Leitlinien; Teil 2: Zielbilder der Schwerpunktthemen". IT-Planungsrat, 26. März 2025. https://www.it-planungsrat.de/fileadmin/it-planungsrat/der-it-planungsrat/foederale_digitalstrategie/250513_IT_PLR_Foerderale_Digitalstrategie_Zukunftsbild_Leitlinien.pdf.

Izacard, Gautier, und Edouard Grave. „Leveraging Passage Retrieval with Generative Models for Open Domain Question Answering". arXiv:2007.01282. Preprint, arXiv, 3. Februar 2021. https://doi.org/10.48550/arXiv.2007.01282.

Ji, Ziwei, Nayeon Lee, Rita Frieske, u. a. „Survey of Hallucination in Natural Language Generation". *ACM Computing Surveys* 55, Nr. 12 (2023): 1–38. https://doi.org/10.1145/3571730.

JSON-RPC Working Group. „JSON-RPC 2.0 Specification". Jsonrpc.Org, 1. Juli 2010. https://www.jsonrpc.org/specification.

Karpukhin, Vladimir, Barlas Oguz, Sewon Min, u. a. „Dense Passage Retrieval for Open-Domain Question Answering". In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, herausgegeben von Bonnie Webber, Trevor Cohn, Yulan He, und Yang Liu. Association for Computational Linguistics, 2020. https://doi.org/10.18653/v1/2020.emnlp-main.550.

Küchenmeister, Sebastian. „Natural Language Generation: Using AI in Content Creation - TYPO3 the Open Source Enterprise CMS". 28. Februar 2019. https://typo3.com/blog/natural-language-generation-using-ai-in-content-generation.

Lewis, Patrick, Ethan Perez, Aleksandra Piktus, u. a. „Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks". arXiv:2005.11401. Preprint, arXiv, 12. April 2021. https://doi.org/10.48550/arXiv.2005.11401.

Marketing Factory Digital GmbH. *marketing-factory/ai-filemetadata*. PHP. 30. August 2024; Marketing Factory Digital GmbH, released 12. September 2025. https://github.com/marketing-factory/ai-filemetadata.

Materna Information & Communications SE. „Digitale Souveränität stärken: Materna und leistungsstarkes Partnernetzwerk gewinnen Rahmenvertrag".

Materna, 22. Mai 2025. https://www.materna.de/newshub/presse/digitale-souveraenitaet-staerken-materna-und-leistungsstarkes-partnernetzwerk-gewinnen-rahmenvertrag/.

Motylewski, Tymoteusz, Łukasz Uznański, Adam Marcinkowski, und Vaclav Janoch. „EXT:Headless — Headless Main Documentation". TYPO3 Documentation, 17. April 2025. https://docs.typo3.org/p/friendsoftypo3/headless/main/en-us.

Nägler, Frank. „TYPO3 V14: Building a System for Community-Driven AI Integrations". TYPO3, 29. Juni 2025. https://typo3.org/article/typo3-v14-ai-integrations.

NITSAN Technologies. „T3AI: All-in-One TYPO3 AI Extension". 1. Juli 2025. https://github.com/nitsan-technologies/ns_t3ai?tab=readme-ov-file.

OpenID Foundation. „OpenID Connect Core 1.0". Openid.Net, 15. Dezember 2023. https://openid.net/specs/openid-connect-core-1_0.html.

PHP-FIG. „PSR-3: Logger Interface - PHP-FIG". PHP-FIG. Zugegriffen 19. September 2025. https://www.php-fig.org/psr/psr-3/.

Raffel, Colin, Noam Shazeer, Adam Roberts, u. a. „Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer". arXiv:1910.10683. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. Preprint, arXiv, 19. September 2023. https://doi.org/10.48550/arXiv.1910.10683.

Reimers, Nils, und Iryna Gurevych. „Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks". arXiv:1908.10084. Preprint, arXiv, 27. August 2019. https://doi.org/10.48550/arXiv.1908.10084.

Schema.org. „Schema.org". Zugegriffen 11. September 2025. https://schema.org/.

Schnabel, Manuel. „Extension AI SEO-Helper — AI SEO-Helper 0.7 Documentation". 2024. https://docs.typo3.org/p/passionweb/ai-seo-helper/0.7/en-us/.

Sporny, Manu, Dave Longley, Gregg Kellogg, Markus Lanthaler, Pierre-Antoine Champin, und Niklas Lindström. „JSON-LD 1.1". W3C Technical Reports, 16. Juli 2020. https://www.w3.org/TR/json-ld11/.

TYPO3 contributors. „Backend User Groups — TYPO3 Explained Main Documentation". 19. September 2025. https://docs.typo3.org/m/typo3/reference-coreapi/main/en-

us/Administration/UserManagement/Groups/Index.html#backend-user-groups.

TYPO3 contributors. „Log Writers — TYPO3 Explained Main Documentation". 19. September 2025. https://docs.typo3.org/m/typo3/reference-coreapi/main/en-us/ApiOverview/Logging/Writers/.

TYPO3 contributors. „Permissions Management — TYPO3 Explained Main Documentation". 19. September 2025. https://docs.typo3.org/m/typo3/reference-coreapi/main/en-us/Administration/PermissionsManagement/.

TYPO3 contributors. „Roles — TYPO3 Explained Main Documentation". 19. September 2025. https://docs.typo3.org/m/typo3/reference-coreapi/main/en-us/ApiOverview/Backend/AccessControl/Roles/.

TYPO3 contributors. „The Logging Framework (Developer Guide)". TYPO3 Documentation, 12. September 2025. https://docs.typo3.org/m/typo3/reference-coreapi/main/en-us/ApiOverview/Logging.

TYPO3 contributors. „TYPO3 Workspaces — Workspaces Main Documentation". 22. August 2025. https://docs.typo3.org/c/typo3/cms-workspaces/main/en-us.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, u. a. „Attention Is All You Need". arXiv:1706.03762. Preprint, arXiv, 2. August 2023. https://doi.org/10.48550/arXiv.1706.03762.

Web Hypertext Application Technology Working Group (WHATWG). *HTML Standard*. Spezifikation. Version Living Standard. WHATWG, living standard. https://html.spec.whatwg.org/.

World Wide Web Consortium (W3C). „RDF 1.1 Concepts and Abstract Syntax". W3C Technical Reports, 25. Februar 2014. https://www.w3.org/TR/rdf11-concepts/.

World Wide Web Consortium (W3C). „Web Content Accessibility Guidelines (WCAG) 2.2". W3C Technical Reports, 12. Dezember 2024. https://www.w3.org/TR/WCAG22/.

Anhang

Anhang A: Fragebogen (Volltext)

Titel: Umfrage zur Einschätzung von Einsatzpotenzialen des Model Context Protocols (MCP) im öffentlichen CMS-Umfeld.

Kurzintro:

Im Rahmen einer Bachelorarbeit an der Technischen Hochschule Brandenburg werden Einschätzungen zur Integration des Model Context Protocols in Content-Management-Systeme erhoben, am Beispiel TYPO3 und mit Blick auf den öffentlichen Sektor. Die Fragen erfassen Erfahrungen, Potenziale, Anforderungen und Hürden aus Praxis- und Technikperspektive. Eine anonyme Teilnahme ist möglich; eine namentliche Nennung erfolgt nur nach ausdrücklicher Zustimmung in der Abschlussfrage.

Hinweis: Die folgenden Fragen sind dem Originalformular entnommen; Format und Reihenfolge wurden beibehalten. Einleitungstext und MCP-Hinweis sind auszugsweise in Textform wiedergegeben.

I. Allgemeiner Hintergrund

A.1 E-Mail-Adresse

Antwortformat: Freitext

Hinweis: Diente der Kontaktaufnahme. Die wissenschaftliche Auswertung erfolgt anonym. E-Mail-Adressen werden in der Arbeit nicht veröffentlicht.

A.2 Arbeitsbereich

Wortlaut: „In welchem Bereich arbeiten Sie derzeit (z. B. öffentlicher Dienst, Agentur, CMS-Entwicklung, IT-Beratung)"

Antwortformat: Freitext.

A.3 Vorerfahrungen

Wortlaut: „Haben Sie bereits Erfahrungen mit TYPO3, anderen CMS-Systemen oder KI-basierten Systemen gesammelt"

Antwortformat: Einzelauswahl Ja/Nein.

II. Potenzial und Relevanz von KI im CMS-Umfeld

A.4 Rolle von KI

Wortlaut: „Welche Rolle spielt Ihrer Meinung nach KI-gestützte Inhaltsnutzung aktuell oder zukünftig in öffentlichen Webanwendungen"

Antwortformat: Freitext.

A.5 Potenziale

Wortlaut: „Wo sehen Sie die größten Potenziale für den Einsatz von KI in Verbindung mit CMS-Systemen"

Antwortformat: Freitext.

A.6 Hürden

Wortlaut: „Gibt es technische oder organisatorische Hürden, die eine solche Integration aktuell erschweren"

Antwortformat: Freitext.

III. Einschätzung des Model Context Protocols (MCP)

Hinweisblock im Formular:

Kurzbeschreibung des MCP als offenes Protokoll zur strukturierten Kontextbereitstellung für KI-Systeme, Verortung im CMS-Kontext, Verweis auf Diskussionen z. B. in Drupal sowie den Fokus dieser Arbeit auf TYPO3.

A.7 Grundsätzliche Bewertung

Wortlaut Teil a): „Halten Sie den Ansatz für grundsätzlich sinnvoll"

Antwortformat: Einzelauswahl Ja/Nein.

A.8 Chancen und Herausforderungen der Umsetzung

Wortlaut Teil b): „Welche Herausforderungen oder Chancen sehen Sie bei der praktischen Umsetzung in CMS-Systemen wie TYPO3"

Antwortformat: Freitext.

Zusatzfrage im selben Block: „Was könnte aus Ihrer Sicht MCP von bestehenden Alternativen (z. B. Google Agent-to-Agent, proprietäre APIs) unterscheiden oder besser machen"

Antwortformat: Freitext.

A.9 Vorteile eines Standardprotokolls

Wortlaut: „Welche Vorteile könnte ein standardisiertes Protokoll wie MCP für Redaktionen, Entwicklerinnen und Nutzerinnen bieten"

Antwortformat: Freitext.

IV. Relevanz im öffentlichen Sektor

A.10 Realismus und Nutzen im öffentlichen Umfeld

Wortlaut: „Halten Sie eine MCP-basierte Integration in öffentlichen CMS-Projekten (z. B. Bund, Kommunen) für realistisch oder hilfreich? Wo könnte diese Art der Integration den höchsten Nutzen erzielen"

Antwortformat: Freitext.

A.11 Anforderungen im öffentlichen Umfeld

Wortlaut: „Welche typischen Anforderungen im öffentlichen Umfeld müssten berücksichtigt werden (z. B. Datenschutz, Erweiterbarkeit, Schnittstellenstandards)"

Antwortformat: Freitext.

V. Feedback zur Idee / Konzeption

A.12 Muss-Kriterien des Konzepts

Wortlaut: „Welche Anforderungen sollte ein Konzept für MCP-Integration aus Ihrer Sicht auf jeden Fall erfüllen"

Antwortformat: Freitext.

A.13 Komponentenempfehlungen für einen Prototyp

Wortlaut: „Haben Sie eine Empfehlung, welche technischen Komponenten (z. B. Schnittstellenformate, Architekturansätze) sich für einen Prototyp eignen würden"

Antwortformat: Freitext.

VI. Abschluss

A.14 Einverständnis zur namentlichen Nennung

Wortlaut: „Wären Sie grundsätzlich bereit, in der Arbeit namentlich erwähnt zu werden"

Antwortformat: Einzelauswahl Ja/Nein.

Hinweis: Nennung ausschließlich bei Zustimmung und nur mit expliziter Freigabe.

A.15 Offene Schlussfrage

Wortlaut: „Gibt es weitere Punkte, die Sie in Bezug auf MCP oder KI in CMS-Kontexten für wichtig halten"

Antwortformat: Freitext.