

# Einführung zum Thema – Umgang mit Legacy-Software...

Workshop des Teams „Wirtschaftsinformatik – Systementwicklung“ – Berlin, 27. April 2026

*Andreas Schmietendorf*

HWR Berlin & Otto-von-Guericke-Universität Magdeburg

# Agenda - Workshop

## *Vorstellungsrunde*

*Andreas Schmietendorf, HWR Berlin & OvG-Universität Magdeburg*

Einführung zum Thema – Umgang mit Legacy-Software

*Tobias Leicher, IBM*

Legacy Software, ist das Kunst oder kann das weg ...

*Ben Rymar, HWR Berlin*

Moderation der Frage- und Diskussionsrunde

# Agenda - Vortrag

- Hintergründe Legacy/Mainframe
- „Silver Bullet“ KI zur Legacy-Transformation
- Legacy Modernisierung

# Hintergründe Legacy/Mainframe

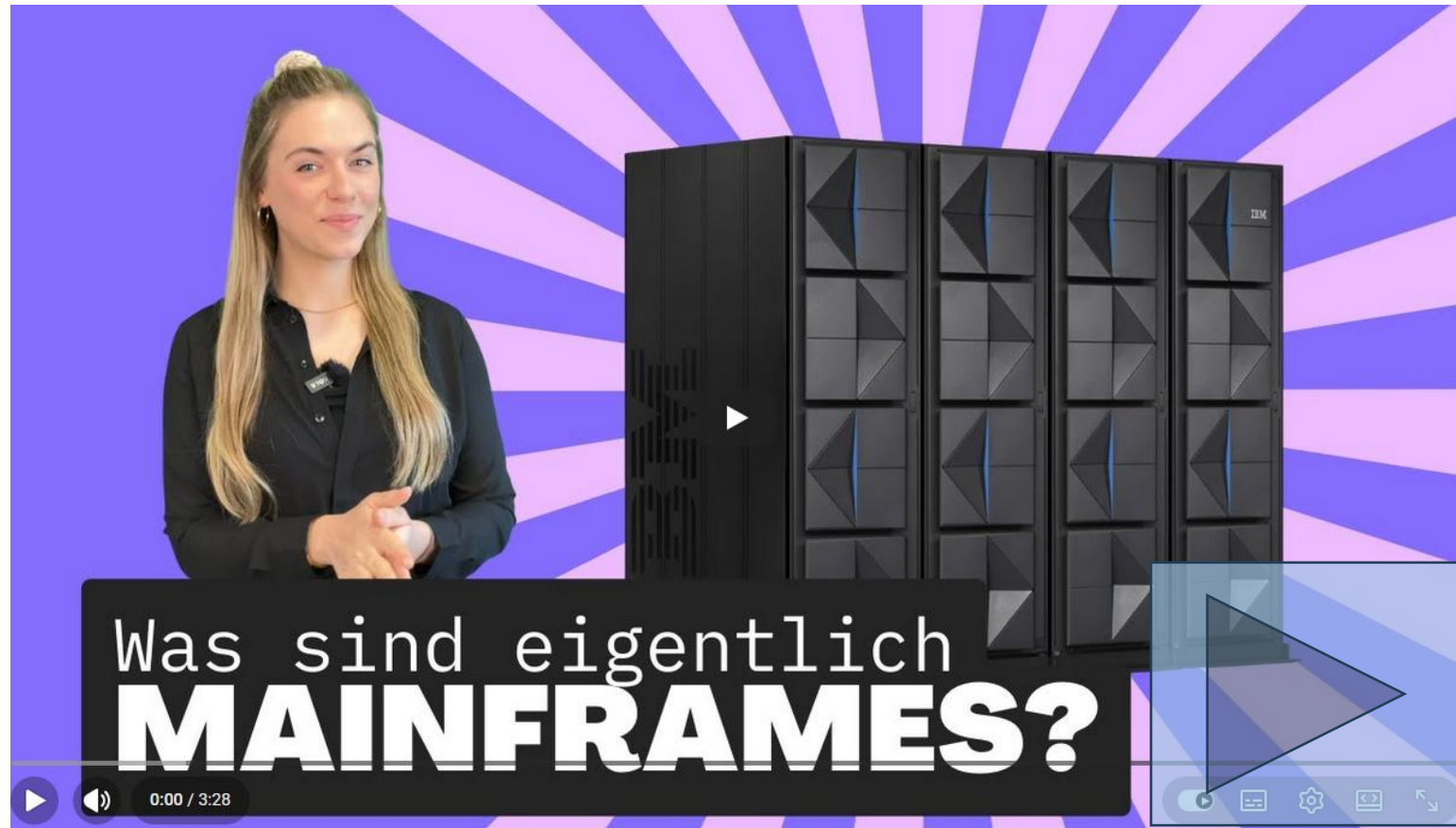
# Definition Legacy System

„Ein **Legacy System** oder eine **Legacy Software** ist ein Altsystem, welches historisch gewachsen ist und nach wie vor in Verwendung ist. In vielen Unternehmen gibt es solche Systeme, die über lange Zeit genutzt, betrieben und gewartet werden. Während dieser Begriff im Deutschen häufig negativ belegt ist (»**Altlasten**«), hat er im Englischen eine durchaus positive Bedeutung: »Vermächtnis oder »Erbe« sind Begriffe, die häufig etwas mit einem hohen Wert beschreiben.“

Quelle des Zitats: Webseite des Fraunhofer IESE zum Thema Was ist ein Legacy System?, April 2026

<https://www.iese.fraunhofer.de/de/leistungen/legacy-systeme.html>

# Begriff des Mainframes



Quelle der Abbildung: Was sind eigentlich Mainframes,  
<https://www.youtube.com/watch?v= YZj25ulEk4>, letzter Zugriff: 20.01.2026

# Legacy Transformation

- Welche Rolle können in der Vergangenheit bewährte IT-Lösungen (legacy systems) in der Zukunft spielen – Aspekt des Investitionsschutzes?
- In welcher Weise sind diese Lösungen an neue Rahmenbedingungen (diversifizierte Ursachen) anzupassen bzw. durch ggf. neue Lösungen zu ersetzen?
- Welche Veränderungen (Organisation, Kultur, Architektur, ...) gilt es neben einer technischen Betrachtung zu berücksichtigen?
- Wie stark wird eine Legacy-Transformation durch angehäufte technischen Schulden (Architekturerosion) behindert?
- Lassen sich agile orientierte domänenspezifische Integrationsarchitekturen umsetzen (Geschäftsprozesse- und daten bzw. Microservice/Web-APIs)?
- In welcher Weise kann der Einsatz von KI die Transformation von Altsystemen unterstützen und wo liegen ggf. Probleme?



# „Silver Bullet“ KI zur Legacy Transformation

# Beispiele für Möglichkeiten mit KI

- Analyse und Dokumentation von Altsystemen (Review-Unterstützung):
  - Modellbasierte Darstellung/Abstraktion von Abhängigkeiten (Komponenten und deren Kopplung).
  - Identifikation von totem oder redundantem Code (cut & paste).
- KI-gestütztes Software-Reengineering (Reverse-Engineering, Refactoring, ...):
  - Quellcodetransformation (Cobol nach Java, C nach Rust, ...).
  - Bereitstellung von API-Spezifikationen bzw. –Kapseln.
- Qualitätssicherung und Testen mit Hilfe von KI:
  - Testfall und Testdatengenerierung mittels KI.
  - KI-gestützte Regressionstests.
- Implizites Wissen extrahieren (vgl. Big Data):
  - Diversifizierte Datenquellen - Dokus, Tickets, E-Mails, ...
  - Idee eines (Agentic RAG-) Legacy-Chatbots.

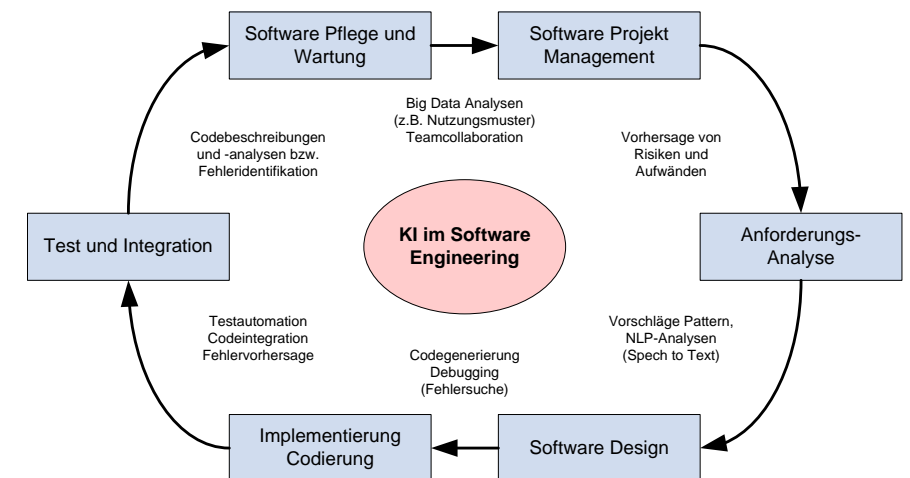


Abb. In Anlehnung an.: Barenkamp, M.; Rebstadt, J.; Thomas, O. (2020): Applications of AI in classical software engineering. In: AI Perspect 2 (1), S. 1–15. DOI: 10.1186/s42467-020-00005-4, letzter Abruf Feb. 2023

# Beispiele für Probleme mit KI

- Aspekte der Qualität - „Garbage in, garbage out“:
  - Unzureichende Dokumentation existierender Fachlogiken – fehlende „Know How“ Träger.
  - Halluzinationen beim Software Reengineering berücksichtigen.
  - Domänenspezifische Architektur mittels KI nur bedingt extrahierbar – Problem Ist/Soll-Architektur!
- Risiken im Bezug auf die Sicherheit- und Compliance:
  - Einsatz sensibler Daten in Trainings- oder Analyseprozessen (Vertrauen).
  - Unklare Datenverarbeitung von KI-Angeboten aus der Cloud (XAI).
  - Umgang mit regulatorischen Anforderungen (z. B. DSGVO, AI Act, BaFin).
- Veränderte Kompetenzanforderungen (ggf. Know-how-Verlust):
  - KI-Ergebnisse sind qualitativ zu sichern (Gefahr, dass Mitarbeitende KI „blind“ folgen).
  - Sukzessive Reduktion des Systemverständnis in Entwicklungsteams.
  - Bedarf neuer Skills ((Agentic-) Prompt Engineering, Review, Test, Governance – shift left).

# Legacy Modernisierung

# Modernisierung von Legacy-Code

- API-basierte Kapselung des Altsystems bzw. ausgewählter Komponenten
- Rehosting auf eine neue Infrastruktur (ggf. virtualisierte Umgebung)
- Refactoring im Sinne des Investitionsschutzes (Restrukturierung des Quellcodes)
- Reengineering durch Überführung in eine neue Zielsprache (z.B. Cobol nach Java)
- Aufbrechen monolithischer Architekturen (z.B. Einsatz von Microservices)
- ...

# Vielfältige KI-Tools im Coding-Diskurs

- Microsoft GitHub Copilot

<https://github.com/features/copilot>

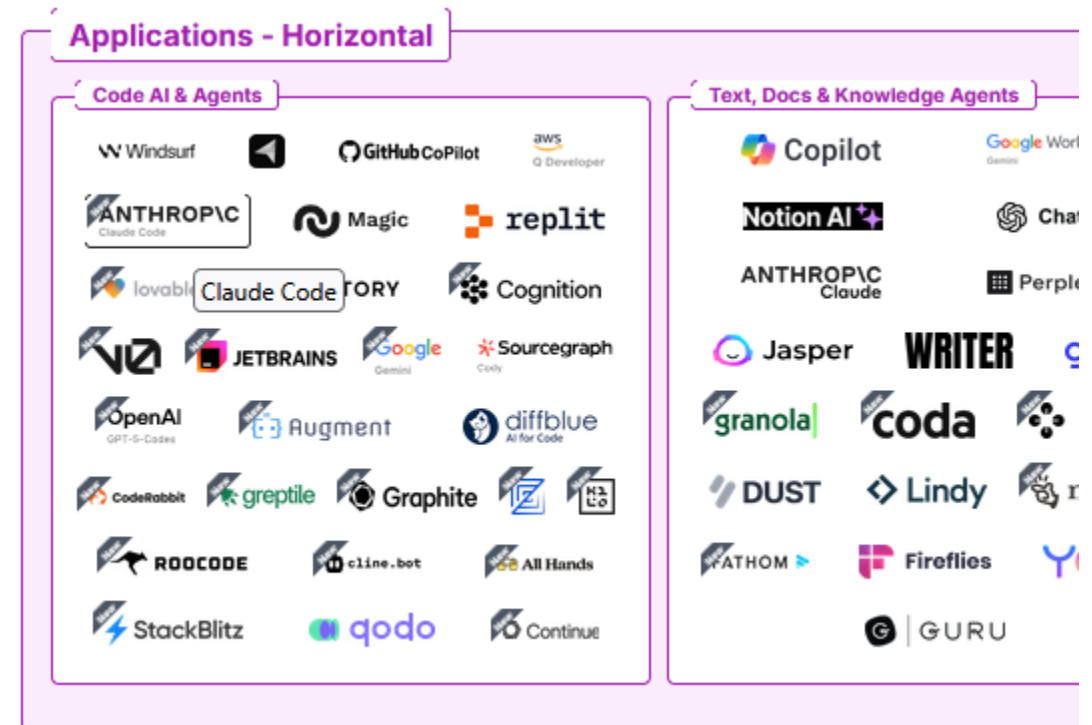
- Cloude Code

<https://claude.com/product/claude-code>

- Tabnine

<https://www.tabnine.com>

- ...



Quelle der rechten Abbildung: The 2025 MAD (ML, AI & Data) Landscape – Ausschnitt (mehr als 2000 KI-Tools!)  
<https://mad.firstmark.com>, letzter Zugriff: 20.01.2026

# Aktuelle Einschätzungen

## Mainframe-Exit für viele Großunternehmen „physisch und finanziell unmöglich“

Marktforscher von Gartner warnen davor, die Fähigkeiten generativer KI-Tools beim Mainframe-Ausstieg zu überschätzen.



(Bild: IBM)



Quelle der linken Abbildung bzw. Artikel: Kannenberg, A.: Mainframe-Exit für viele Großunternehmen..., iX Magazin, 15.04.2026  
<https://www.heise.de/news/Gartner-KI-ist-kein-Wundermittel-fuer-den-Mainframe-Exit-11258947.html>, letzter Zugriff: 26.04.2026

# Weiterführende Informationen

# ceCMG-Tgaung im Mai 2026

- Praxisberichte
- Networking
- Deep-Dive-Sessions
- Round table
- Entwicklungstendenzen
- ...



20  
-  
21  
Mai

benötigte Veränderungen gehen weit über die technischen Veränderungen hinaus und betreffen implizite Organisationsabläufe. [Wer wir sind](#) [Unsere Köpfe](#) [Arbeitskreise](#) [Events](#) [Mitglied werden](#) [Kontakt](#)



## ceCMG Frühjahrstagung 2026 – Mainframe-Modernisierung in der Praxis & Mitgliederversammlung

20.–21. Mai 2026 | Frankfurt am Main

- **Beginn:** 20. Mai, 12:30 Uhr
- **Ende:** 21. Mai, 15:30 Uhr

Mainframe-Modernisierung zählt zu den zentralen Transformationsaufgaben vieler Unternehmen. Unterschiedliche Lösungsansätze – von Rehosting über Refactoring, Code Conversion bis hin zu Neuprogrammierung und hybriden Strategien – versprechen unterschiedliche Vorteile, stoßen jedoch je nach Kontext auch an klare Grenzen.

Beginn: 12:30 Uhr: Bei Interesse besteht die Möglichkeit, an einem Workshop zur Hardwarecompliance bereits um 9:00 Uhr teilzunehmen – bei Interesse an Teilnahme an diesem Workshop bitte in der Anmeldung darauf hinweisen

Zu Beginn der Tagung gibt der **ceCMG-Vorstand** einen strukturierten Überblick über die aktuell relevanten Modernisierungsansätze im Markt. Ziel ist es, die wesentlichen Charakteristika der verschiedenen Optionen einzuordnen und ein gemeinsames Verständnis für deren Einsatzfelder zu schaffen.

In den **anschließenden Deep-Dive-Sessions** werden diese Ansätze anhand konkreter **Praxisbeispiele aus Anwenderunternehmen** vertieft. Kunden berichten offen über ihre Entscheidungswege, Projekterfahrungen, erzielten Ergebnisse sowie über Herausforderungen, Stolpersteine und besonderen Erfolge. Der Fokus liegt dabei ausdrücklich auf realen Erfahrungen und Lessons Learned, nicht auf Marketing- oder Vertriebsbotschaften.

Quelle der rechten Abbildung: <https://cecmg.de/events>, letzter Abruf: 26.04.2026

# Low-Code Publikation

- Low-Code – Agile & fachgetriebene Entwicklung
- Generischer Funktionsumfang von Low-Code-Ansätzen
- Auswahlkriterien für Low-Code Plattformen
- Analyse realer Low-Code Plattformen
  - KNIME
  - Taipy
  - KI-Web-APIs
  - ...
- Wechselwirkungen zwischen KI und Low-Code



Quelle: Schmietendorf, A.; Knuth, M.: Aspekte des Software Engineerings im Diskurs einer Low-Code orientierten Softwareentwicklung, ca. 124 Seiten, Monografie, Logos-Verlag, Berlin, Juni 2024, ISBN 978-3-8325-5733-1

# KI-Prototypen Publikation

- KI unterstütztes Software Engineering
- Software Engineering für fachorientierte KI-Lösungen
- Anonymisierung mit Hilfe von KI
- Experimentelles Prototyping im Diskurs:
  - Professionsforschung (Mediation)
  - Forstwirtschaft (Totholzerkennung)
  - Bahnwirtschaft (Vandalismusbekämpfung)
  - ...
- Rechtsfragen KI-gestützter Analysen (DSGVO, EU AI Act, ...)



Quelle: Hartenstein, S.; Letzel, W. H.; Schmietendorf, A. (Hrsg.): Software Engineering prototypischer KI-Implementierungen im Zusammenhang mit domänenspezifischen Problemen des Projekts TAHAI (TrustAdHocAI), 110 Seiten, Monografie, Logos-Verlag, Berlin, Juni 2024, ISBN 978-3-8325-5906-9