Professionell mit ChatGPT arbeiten

Ein Leitstandard für zuverlässige Ergebnisse

Kapitel 1

Warum ChatGPT oft beeindruckt – und trotzdem unzuverlässig bleibt

Du stellst eine klare Frage.
Die Antwort kommt schnell.
Sie klingt gut.
Sie wirkt durchdacht.
Manchmal sogar brillant.

Und trotzdem bleibt ein Rest Unsicherheit.

War das wirklich korrekt?
Hätte die Antwort auch anders ausfallen können?
Warum klingt sie so sicher, obwohl du weißt, dass sie nicht überprüft wurde?

Viele erleben genau das.

ChatGPT liefert beeindruckende Ergebnisse.
Und gleichzeitig entstehen regelmäßig Fehler.

Manchmal sind es kleine Ungenauigkeiten.
Manchmal sind es erfundene Quellen.
Manchmal sind es logische Brüche.
Und manchmal merkt man es gar nicht.

Das Problem ist nicht, dass ChatGPT schlecht ist.
Im Gegenteil: Für Wissensarbeit ist es ein extrem leistungsfähiges Werkzeug.

Das Problem ist etwas anderes.

Das eigentliche Missverständnis

Viele nutzen ChatGPT so, als würde es „wissen“.
Als hätte es Zugriff auf Wahrheit.
Als würde es prüfen, bevor es antwortet.

Das tut es nicht.

ChatGPT berechnet Wahrscheinlichkeiten.
Es erzeugt Text, der statistisch passt.
Es optimiert auf sprachliche Plausibilität – nicht auf Wahrheit.

Das ist kein Fehler.
Das ist das Design.

Wenn du das nicht weißt, erwartest du Stabilität.
Wenn du es weißt, erkennst du:
Stabilität entsteht nicht automatisch.

Warum gute Prompts nicht reichen

Die naheliegende Reaktion lautet:

„Dann schreibe ich bessere Prompts.“

Das hilft.
Aber es löst das Grundproblem nicht.

Ein besserer Prompt reduziert Varianz.
Er erzeugt klarere Antworten.
Er kann Fehler verringern.

Aber er verändert nicht die Natur des Systems.

Das System bleibt probabilistisch.

Das bedeutet:

- Zwei leicht unterschiedliche Formulierungen können unterschiedliche Antworten erzeugen.
- Kleine Kontextänderungen können den Ton verschieben.
- Unklare Stellen werden gefüllt – auch wenn keine sichere Grundlage existiert.

Das ist keine Schwäche.
Es ist die Funktionsweise.

Ein einfaches Beispiel

Frage 1:
„Erkläre mir den AI Act in einfachen Worten.“

Frage 2:
„Erkläre mir den AI Act juristisch präzise.“

Du bekommst zwei unterschiedliche Antworten.
Beide können richtig sein.
Beide können unterschiedliche Details betonen.

Jetzt veränderst du nur einen Nebensatz.
Oder lässt eine Information weg.
Oder stellst eine Anschlussfrage.

Das Ergebnis verschiebt sich.

Nicht dramatisch.
Aber spürbar.

Wenn du das System wie ein Werkzeug behandelst, das deterministisch arbeitet, entsteht Frustration.

Wenn du es als probabilistisches Sprachsystem verstehst, entsteht Klarheit.

Der entscheidende Punkt

Unzuverlässigkeit entsteht nicht durch „dumme KI“.

Sie entsteht durch fehlende Struktur im Umgang mit ihr.

ChatGPT antwortet immer.
Es signalisiert Sicherheit.
Es füllt Lücken.

Ohne klare Arbeitsweise entsteht:

- implizite Annahme
- unbemerkte Kontextverschiebung
- scheinbare Sicherheit

Das fühlt sich wie Präzision an.
Ist es aber nicht automatisch.

Erste Orientierung

Wenn Prompts keine Architektur sind,
dann braucht professionelle Nutzung etwas anderes.

Nicht mehr Kreativität im Prompt.
Nicht mehr Tricks.
Sondern Struktur.

Struktur bedeutet:

- Klar definierte Ziele
- Klare Zustände
- Klare Grenzen
- Klare Stop-Regeln

Wie das konkret aussieht, klären wir im nächsten Kapitel.

Zusammenfassung Kapitel 1

- ChatGPT ist leistungsfähig, aber probabilistisch.
- Gute Prompts verbessern Ergebnisse, verändern aber nicht die Natur des Systems.
- Unzuverlässigkeit entsteht durch fehlende Struktur.
- Professionelle Nutzung beginnt nicht beim Prompt, sondern bei der Arbeitsweise.

Kapitel 2

Wie ChatGPT wirklich funktioniert – einfach erklärt

Wenn du professionell mit ChatGPT arbeiten willst, musst du verstehen, womit du es zu tun hast.

Nicht technisch im Detail.
Aber strukturell.

Denn nur was du verstehst, kannst du kontrollieren.

ChatGPT „weiß“ nichts

ChatGPT speichert keine Fakten im klassischen Sinn.
Es greift nicht aktiv auf eine Datenbank zu.
Es prüft nicht, ob eine Aussage stimmt.

Es berechnet Wahrscheinlichkeiten.

Genauer gesagt:

Es sagt voraus, welches Wort mit hoher Wahrscheinlichkeit als Nächstes passt – basierend auf deinem bisherigen Text und dem Kontext.

Das passiert extrem schnell.
Und wirkt deshalb intelligent.

Aber es ist Vorhersage.
Keine Prüfung.
Keine Verifikation.

Warum Antworten so überzeugend klingen

ChatGPT wurde darauf trainiert, sprachlich kohärent zu sein.

Das bedeutet:

- Die Antwort klingt strukturiert.
- Die Sätze sind logisch verbunden.
- Der Ton passt zur Frage.
- Die Form wirkt sicher.

Sprachliche Sicherheit ist aber nicht inhaltliche Sicherheit.

Das ist der zentrale Unterschied.

Wenn ein Text gut klingt, neigen wir dazu, ihn als korrekt wahrzunehmen.

Das System nutzt genau diese Stärke.

Kontext ist alles

ChatGPT arbeitet stark kontextabhängig.

Das bedeutet:

- Jede neue Nachricht verändert den Antwort-Raum.
- Jede implizite Annahme beeinflusst das Ergebnis.
- Jede nicht präzise formulierte Zielsetzung erzeugt Spielraum.

Dieser Spielraum wird gefüllt.

Nicht absichtlich.
Nicht strategisch.
Sondern probabilistisch.

Wenn eine Information fehlt, erzeugt das System eine plausible Ergänzung.

Das kann korrekt sein.
Es kann aber auch eine saubere Halluzination sein.

Was Halluzination wirklich bedeutet

Halluzination heißt nicht, dass das System „lügt“.

Es bedeutet:

Das System erzeugt eine sprachlich plausible Aussage,
für die keine verlässliche Grundlage vorhanden ist.

Das passiert besonders häufig, wenn:

- Konkrete Quellen verlangt werden
- Spezifische Details abgefragt werden
- Der Kontext unklar ist
- Die Frage zu allgemein bleibt

Das System priorisiert Antwortfähigkeit über Zurückhaltung.

Das ist ein Design-Entscheid.

Warum das kein Fehler ist

ChatGPT wurde für Sprachproduktion optimiert.

Für Wissensarbeit ist das extrem wertvoll:

- Strukturieren
- Zusammenfassen
- Ideen generieren
- Perspektiven öffnen
- Texte verbessern

In diesen Bereichen funktioniert es hervorragend.

Probleme entstehen erst, wenn du implizit erwartest,
dass es stabil, prüfend und zustandsbewusst arbeitet.

Das tut es nicht.

Der strukturelle Unterschied

Ein Taschenrechner liefert bei gleicher Eingabe immer dasselbe Ergebnis.

ChatGPT nicht.

Das bedeutet nicht, dass es schlechter ist.
Es ist nur ein anderes System.

Ein Taschenrechner ist deterministisch.
ChatGPT ist probabilistisch.

Wenn du das ignorierst, entsteht Frustration.

Wenn du es verstehst, entsteht Kontrolle.

Zweite Orientierung

Professionelle Nutzung beginnt nicht bei besseren Prompts.

Sie beginnt mit dem Verständnis:

Du arbeitest mit einem System, das Sprache vorhersagt – nicht Wahrheit garantiert.

Wenn du Zuverlässigkeit willst,
musst du Rahmenbedingungen schaffen.

Was diese Rahmenbedingungen sind, klären wir im nächsten Kapitel.

Zusammenfassung Kapitel 2

- ChatGPT berechnet Wahrscheinlichkeiten, es prüft keine Fakten.
- Sprachliche Sicherheit ist nicht gleich inhaltliche Sicherheit.
- Halluzination ist plausible Ergänzung ohne sichere Grundlage.
- Für Wissensarbeit ist das System sehr wertvoll.
- Zuverlässigkeit entsteht nur durch bewusst gesetzte Struktur.

Kapitel 3

Warum Rollen nicht ausreichen

Viele haben gelernt:

„Gib ChatGPT eine Rolle, dann wird es besser."

Zum Beispiel:

- „Du bist ein Marketingexperte."
- „Du bist ein Jurist."
- „Du bist mein Coach."
- „Du bist ein erfahrener Entwickler."

Das funktioniert.
Die Antworten werden fokussierter.
Der Ton wird konsistenter.
Die Perspektive klarer.

Aber es bleibt ein Missverständnis.

Eine Rolle ist keine Struktur.

Was eine Rolle wirklich bewirkt

Eine Rolle beeinflusst den Stil.
Sie beeinflusst die Perspektive.
Sie beeinflusst den Sprachrahmen.

Sie sagt dem System:

„Antworte so, wie diese Figur typischerweise sprechen würde."

Das verbessert Qualität.
Aber es kontrolliert nicht das System.

Es definiert keinen Zustand.
Es definiert keine Grenzen.
Es definiert keine Stop-Regeln.

Ein Beispiel

Prompt:

„Du bist ein erfahrener Strategieberater. Analysiere dieses Geschäftsmodell."

Die Antwort wirkt strukturiert.
Sie klingt fundiert.
Sie hat Absätze, Argumente, Bewertung.

Jetzt veränderst du die Rolle:

„Du bist ein kritischer Investor."

Die Analyse verschiebt sich.

Beide Antworten können sinnvoll sein.
Beide können schlüssig wirken.
Beide können plausibel sein.

Was fehlt, ist ein stabiler Referenzrahmen.

Rolle gehört in den Modus – aber nicht als Ersatz

Eine Rolle ist kein Arbeitsrahmen.

Aber sie hat einen sinnvollen Platz.

Verwende Rollen für:

- Wortwahl
- Tonlage
- Tiefe
- Zielgruppenniveau

Nicht für:

- Struktur
- Begrenzung
- Stop-Regeln

Ein professioneller Modus kann deshalb eine Rolle enthalten.

Beispiel:

Analyse-Modus
Rolle: Sachlicher Fachexperte
Ziel: Beschreibung
Verboten: Bewertung

Die Rolle beeinflusst die Sprache.
Der Modus steuert die Arbeit.

Diese Trennung verhindert Verwechslung.

Rollen erzeugen Perspektive – keine Kontrolle

Eine Rolle sagt:

„Aus welcher Sicht antwortest du?“

Sie sagt nicht:

- Wann darf das System spekulieren?
- Wann soll es Unsicherheit markieren?
- Wann soll es stoppen?
- Welche Annahmen sind erlaubt?
- Welche nicht?

Das System füllt weiterhin Lücken.

Nur jetzt im Stil der gewählten Rolle.

Das Kernproblem

Wenn du glaubst, eine Rolle löse das Zuverlässigkeitsproblem,
überschätzt du ihre Wirkung.

Rollen steuern Ton und Fokus.
Sie steuern nicht Verantwortung.

Professionelles Arbeiten braucht mehr als eine Perspektive.

Es braucht:

- explizite Zieldefinition
- explizite Begrenzung
- explizite Annahmen
- explizite Stop-Punkte

Das leistet eine Rolle nicht.

Warum das oft übersehen wird

Rollen fühlen sich wie Kontrolle an.

Du sagst dem System, wer es sein soll.
Die Antwort klingt danach.
Das erzeugt Vertrauen.

Dieses Vertrauen ist teilweise berechtigt.
Aber es ist kein Sicherheitsmechanismus.

Es ist ein Stilmechanismus.

Übergang

Wenn Rollen nicht reichen,
braucht es etwas anderes.

Nicht eine bessere Rolle.
Nicht mehr Kontext.
Sondern eine andere Arbeitsweise.

Diese Arbeitsweise basiert nicht auf Figuren,
sondern auf Zuständen.

Was das bedeutet, klären wir im nächsten Kapitel.

Zusammenfassung Kapitel 3

- Rollen verbessern Stil und Perspektive.
- Rollen schaffen keine strukturelle Kontrolle.
- Rollen definieren keine Grenzen, Annahmen oder Stop-Regeln.
- Professionelle Nutzung braucht mehr als eine gut gewählte Rolle.

Kapitel 4

Zustände ermöglichen konsistentes Arbeiten – nicht nur bessere Prompts

Wenn du bisher mit ChatGPT gearbeitet hast, kennst du das Muster:

Du formulierst einen guten Prompt.
Du bekommst eine gute Antwort.
Beim nächsten Thema beginnst du wieder von vorne.

Wieder erklären.
Wieder eingrenzen.
Wieder präzisieren.

Das wirkt wie Kontrolle.
Ist aber nur Wiederholung.

Ein einzelner Prompt löst kein Strukturproblem.
Er löst nur eine einzelne Situation.

Professionelles Arbeiten braucht mehr als gute Einzelanweisungen.

Es braucht Kontinuität.

Ein Prompt ist ein Moment.

Ein Zustand ist ein Rahmen.

Ein Prompt wirkt nur für eine Antwort.

Ein Zustand wirkt über viele Fragen hinweg.

Das ist der entscheidende Unterschied.

Wenn du nur mit Prompts arbeitest,
baust du jedes Mal neu auf.

Wenn du mit Zuständen arbeitest,
arbeitest du innerhalb eines festgelegten Rahmens.

Dieser Rahmen bleibt bestehen,
bis du ihn bewusst änderst.

Warum das wichtig ist

ChatGPT ist kontextsensitiv.

Das bedeutet:

Jede neue Nachricht verändert den Antwort-Raum.

Wenn kein klarer Arbeitszustand definiert ist,
verschiebt sich der Rahmen unbemerkt.

Heute analysierst du.
Im nächsten Schritt bewertest du.
Dann entsteht eine Empfehlung.
Dann eine Annahme.

Ohne Übergang.
Ohne bewusste Entscheidung.

Ein dauerhafter Zustand verhindert diese Drift.

Was ein Modus wirklich ist

Ein Modus ist kein längerer Prompt.

Ein Modus ist ein benannter Arbeitsrahmen,
der über mehrere Interaktionen hinweg gilt.

Zum Beispiel:

Analyse-Modus
Kritik-Modus
Ideen-Modus
Validierungs-Modus

Ein Modus definiert:

- Ziel der aktuellen Phase
- Erlaubte Handlungen
- Nicht erlaubte Handlungen
- Umgang mit Unsicherheit

Und dieser Rahmen bleibt aktiv,
bis du ihn änderst.

Du musst ihn nicht jedes Mal neu schreiben.
Du wechselst ihn bewusst.

Beispiel

Ohne Modus:

„Analysiere das Geschäftsmodell.“
„Was hältst du davon?“
„Wie könnte man es verbessern?“

Die Phasen vermischen sich.
Analyse, Bewertung und Optimierung laufen ineinander.

Mit Modus:

Analyse-Modus aktiv.
Nur Beschreibung. Keine Bewertung.

Dann bewusster Wechsel:
Kritik-Modus aktiv.
Nur Schwächen. Keine Lösungen.

Dann Wechsel:
Optimierungs-Modus aktiv.
Nur Verbesserungsvorschläge.

Der Unterschied liegt nicht im Prompt.
Er liegt in der Phasendisziplin.

Zustände reduzieren Wiederholung

Wenn du jedes Mal alles neu formulierst,
arbeitest du gegen das System.

Wenn du Zustände definierst und beibehältst,
arbeitest du mit dem System.

Das spart:

- Erklärungsaufwand
- Missverständnisse
- Kontextverschiebung
- unnötige Variabilität

Zustände müssen gespeichert werden

Ein Modus wirkt nur dann wirklich stabil,
wenn er wiederverwendbar ist.

Wenn du ihn jedes Mal neu formulierst,
arbeitest du wieder im Einzel-Prompt-Denken.

Deshalb:

Speichere deine Arbeitsmodi.

Nutze Custom Instructions oder feste Modus-Vorlagen.

So kannst du schreiben:

„Analyse-Modus aktiv."

Und der definierte Rahmen gilt.

Das reduziert:

- Wiederholung
- Inkonsistenz
- Vergessen von Regeln

Persistente Modi sind der Übergang
von Nutzung zu Systemarbeit.

Der eigentliche Gewinn

Ein einzelner Prompt verbessert eine Antwort.

Ein Zustand verbessert den gesamten Dialog.

Das ist der Schritt vom „Fragen stellen"
zum strukturierten Arbeiten.

Übergang

Wenn Zustände der Rahmen sind,
stellt sich die nächste Frage:

Wie baut man einen Modus so auf,
dass er klar, stabil und praktikabel bleibt?

Das klären wir im nächsten Kapitel.

Zusammenfassung Kapitel 4

- Ein Prompt wirkt nur für eine einzelne Antwort.
- Ein Zustand wirkt über viele Interaktionen hinweg.
- Ein Modus ist ein benannter, persistenter Arbeitsrahmen.
- Zustände verhindern Drift zwischen Analyse, Bewertung und Entscheidung.
- Struktur entsteht durch bewusste Modus-Wechsel, nicht durch längere Prompts.

Kapitel 5

Der Modus-Baukasten

Ein Modus ist ein dauerhafter Arbeitsrahmen.

Damit er stabil funktioniert, reicht ein Name nicht.

„Analyse-Modus“ klingt gut.
Aber ohne klare Definition bleibt er unscharf.

Ein professioneller Modus besteht immer aus denselben Bausteinen.

Nicht mehr.
Nicht weniger.

1. Ziel

Definiere, was in diesem Modus erreicht werden soll.

Beispiel:

- Nur beschreiben
- Nur Schwächen identifizieren
- Nur Hypothesen sammeln
- Nur strukturieren

Ein Modus ohne klares Ziel driftet.

2. Erlaubte Handlungen

Lege fest, was das System tun darf.

Zum Beispiel:

- Zusammenfassen
- Strukturieren
- Fragen stellen
- Perspektiven aufzeigen

Ohne klare Erlaubnis erweitert das System den Handlungsspielraum selbst.

3. Nicht erlaubte Handlungen

Das ist der wichtigste Teil.

Definiere klar, was nicht passieren darf.

Zum Beispiel:

- Keine Bewertung
- Keine neuen Annahmen
- Keine Ergänzung fehlender Daten
- Keine Handlungsempfehlung

Wenn du diesen Teil weglässt, entsteht implizite Erweiterung.

4. Umgang mit Unsicherheit

Lege fest, wie Unsicherheit behandelt wird.

Zum Beispiel:

- Unsicherheit explizit markieren
- Bei fehlender Information stoppen
- Rückfragen stellen statt ergänzen

Hier entstehen die meisten Halluzinationen.
Nicht durch schlechte Absicht, sondern durch fehlende Begrenzung.

5. Stop-Regel

Definiere, wann das System abbrechen soll.

Zum Beispiel:

- Wenn Daten fehlen
- Wenn mehrere Interpretationen möglich sind
- Wenn Annahmen notwendig wären

Eine Stop-Regel verhindert Überdehnung.

Warum dieser Baukasten funktioniert

ChatGPT reagiert auf Wahrscheinlichkeiten.

Wenn du Ziel, Grenzen und Stop-Regeln definierst,
verengst du den Wahrscheinlichkeitsraum.

Das erzeugt Stabilität.

Nicht Perfektion.
Aber deutlich weniger Drift.

Ein einfacher Modus in Praxisform

Analyse-Modus:

Ziel: Sachliche Beschreibung.
Erlaubt: Strukturieren, Zusammenfassen.
Nicht erlaubt: Bewertung, Optimierung, Spekulation.
Unsicherheit: Kennzeichnen.
Stop: Bei fehlenden Daten.

Mehr braucht es nicht.

Ein klarer Rahmen.
Kein langer Text.

Der entscheidende Unterschied

Ohne Baukasten:

„Sei bitte analytisch."

Mit Baukasten:

Klare Zieldefinition.
Klare Grenzen.
Klare Stop-Regeln.

Der zweite Ansatz reduziert implizites Weiterdenken erheblich.

Übergang

Modi schaffen Struktur.

Aber Struktur allein verhindert noch keine Halluzination.

Im nächsten Kapitel geht es um die Mechanismen,
mit denen du Halluzination aktiv reduzierst.

Zusammenfassung Kapitel 5

- Ein Modus braucht fünf Bausteine: Ziel, Erlaubtes, Verbotenes, Unsicherheit, Stop-Regel.
- Klare Grenzen reduzieren implizite Erweiterung.
- Ein Modus ist kurz, aber präzise.
- Stabilität entsteht durch Begrenzung des Handlungsspielraums
- **Ohne gespeicherte Modi bleibt professionelle Nutzung dauerhaft anstrengend**

Kapitel 6

Halluzination verstehen – und strukturell abschalten

Halluzination ist kein Zufall.
Sie ist eine logische Folge der Funktionsweise von Sprachmodellen.

Wenn Information fehlt,
erzeugt das System eine plausible Ergänzung.

Nicht, weil es täuschen will.
Sondern weil es auf Antwortfähigkeit optimiert ist.

Ein LLM ist darauf trainiert,
kohärent weiterzuschreiben.

Nicht darauf, Unsicherheit zu priorisieren.

Wie Halluzination entsteht

Drei typische Auslöser:

1. Fehlende Information

Wenn Daten unvollständig sind,
wird der wahrscheinlichste Anschluss erzeugt.

Das kann korrekt sein.
Oder plausibel falsch.

2. Zu großer Interpretationsraum

Unklare Fragen erzeugen breite Antwortmöglichkeiten.

Je größer der Raum, desto größer die Varianz.

Varianz bedeutet Instabilität.

3. Implizite Erwartung von Vollständigkeit

Das System wurde trainiert zu antworten.
Nicht zu schweigen.

Wenn eine Antwort erwartet wird,
wird sie erzeugt.

Auch dann, wenn die Grundlage schwach ist.

Warum das gefährlich wird

Sprachliche Sicherheit wirkt wie inhaltliche Sicherheit.

Eine sauber formulierte, strukturierte Antwort
erzeugt Vertrauen.

Wenn Annahmen nicht sichtbar sind,
wirken sie wie Fakten.

Das ist der kritische Punkt.

Nicht die Halluzination selbst.
Sondern ihre Unsichtbarkeit.

Halluzination wird nicht durch bessere Prompts gelöst

Viele versuchen:

„Bitte keine Annahmen."
„Bitte nur geprüfte Fakten."
„Bitte nur sichere Quellen."

Das reduziert Risiko – einmal.

Beim nächsten Kontextwechsel beginnt das Problem neu.

Solange der Mechanismus aktiv ist,
bleibt das Risiko aktiv.

Professionelle Nutzung schaltet Halluzination nicht situativ aus.
Sie begrenzt systematisch den Raum, in dem sie entstehen kann.

Der strukturelle Hebel

Halluzination entsteht, wenn:

- Ergänzung erlaubt ist
- Unsicherheit nicht markiert werden muss
- Stop-Regeln fehlen

Deshalb gehören Anti-Halluzinationsregeln in den Modus.

Nicht als Zusatz.
Sondern als Grundbestandteil.

Ein stabiler Arbeitsmodus enthält:

- Pflicht zur Kennzeichnung von Unsicherheit
- Verbot impliziter Ergänzungen
- Trennung von Fakten und Interpretation
- Klare Stop-Regel bei fehlender Grundlage

Damit verändert sich nicht nur eine Antwort.
Sondern der gesamte Dialograhmen.

Was sich dadurch ändert

Ohne integrierte Regeln:

Antwort → Korrektur → Präzisierung → Nachschärfung.

Mit integrierten Regeln:

Antwort mit markierter Unsicherheit → klare Weiterarbeit.

Du reparierst weniger.
Du kontrollierst mehr.
Du vertraust dem Prozess.

Das ist schneller.
Und zuverlässiger.

Wichtig

LLMs sind extrem leistungsfähig für Wissensarbeit.

Sie strukturieren, formulieren, beschleunigen.

Halluzination ist kein Zeichen von Schwäche.
Sondern ein Nebenprodukt von Sprachvorhersage.

Wenn du sie nicht begrenzt,
arbeitet das System implizit.

Wenn du sie strukturell einhegst,
arbeitest du kontrolliert.

Übergang

Modi schaffen Rahmen.
Integrierte Anti-Halluzinationsregeln schaffen Stabilität.

Jetzt fehlt nur noch ein Element:

Wie stellst du sicher,
dass dieser Rahmen dauerhaft eingehalten wird?

Darum geht es im nächsten Kapitel.

Zusammenfassung Kapitel 6

- Halluzination entsteht durch Lückenergänzung in probabilistischen Systemen.
- Sprachliche Sicherheit ist nicht gleich inhaltliche Sicherheit.
- Einzelne Anti-Halluzinationshinweise reichen nicht.
- Die Regeln müssen Teil des Modus sein.
- Dauerhafte Struktur macht Arbeit schneller und verlässlicher.

Kapitel 7

Governance: Wie du Struktur dauerhaft erhältst

Ein Modus schafft Struktur.

Anti-Halluzinationsregeln schaffen Stabilität.

Aber beides reicht nicht,
wenn der Rahmen unbemerkt wieder verloren geht.

Struktur verschwindet schleichend.

Nicht abrupt.
Sondern durch kleine Übergänge.

Eine zusätzliche Frage.
Eine neue Perspektive.
Ein spontaner Gedanke.

Ohne klare Governance driftet der Modus.

Was Drift bedeutet

Drift entsteht, wenn:

- Analyse in Bewertung übergeht
- Bewertung in Empfehlung
- Empfehlung in Annahme
- Annahme in scheinbare Tatsache

Ohne bewussten Wechsel.

Das System folgt dem Gespräch.
Wenn du nicht steuerst,
verschiebt sich der Zustand.

Governance ist keine Bürokratie

Governance bedeutet nicht Kontrolle aus Misstrauen.

Governance bedeutet:

Den Arbeitsrahmen aktiv führen.

Nicht das System arbeiten lassen.
Sondern das System lenken.

Drei Grundregeln für stabile Modi

1. Benenne den aktiven Modus

Arbeite nicht implizit.

Wenn du analysierst, sag:
„Analyse-Modus aktiv."

Wenn du kritisierst, sag:
„Kritik-Modus aktiv."

Benennung schafft Bewusstsein.

2. Wechsle Modi bewusst

Vermische keine Phasen.

Beende einen Modus.
Starte den nächsten.

Das verhindert Übergang ohne Entscheidung.

3. Stoppe bei Mehrdeutigkeit

Wenn mehrere Interpretationen möglich sind,
erzwinge Klärung.

Lass nicht weiterarbeiten,
wenn die Grundlage unscharf ist.

Unsicherheit muss sichtbar bleiben.

Einzelschritte mit Bestätigung

Komplexe Aufgaben erzeugen große Wahrscheinlichkeitsräume.

Großer Raum bedeutet große Varianz.

Reduziere diesen Raum durch Einzelschritte.

Statt:

„Erstelle eine komplette Strategie."

Arbeite so:

1. Zieldefinition
2. Analyse
3. Bewertung
4. Empfehlung

Und bestätige jeden Schritt.

„Schritt 1 abgeschlossen. Bestätigen, bevor wir weitergehen."

Das erzeugt:

- Kontrolle
- Fehlerfrüherkennung
- klare Übergänge

Einzelschritt-Arbeit ist langsamer im ersten Moment.

Aber schneller im Gesamtergebnis,
weil Korrekturschleifen entfallen.

Warum das entscheidend ist

Ein LLM ist probabilistisch.

Es folgt dem wahrscheinlichsten Anschluss.

Wenn du keinen klaren Rahmen hältst,
entsteht implizite Autonomie.

Nicht aus Absicht.
Sondern aus Wahrscheinlichkeit.

Governance verhindert das.

Der Unterschied zwischen Nutzung und Führung

Normale Nutzung:

Frage → Antwort → nächste Frage → nächste Antwort.

Professionelle Nutzung:

Modus → Frage → strukturierte Antwort → bewusster Wechsel → nächste Phase.

Im ersten Fall folgt das System dir.

Im zweiten Fall führst du das System.

Der Effekt

Mit Governance:

- Weniger Missverständnisse
- Weniger Korrekturschleifen
- Weniger implizite Annahmen
- Mehr Klarheit
- Mehr Geschwindigkeit

Nicht durch Kontrolle im Detail.
Sondern durch Klarheit im Rahmen.

Übergang

Du hast jetzt:

- Verständnis für die Funktionsweise
- Klarheit über Zustände
- Einen Modus-Baukasten
- Integrierte Anti-Halluzinationsregeln
- Governance zur Stabilisierung

Im nächsten Kapitel geht es darum,
wie diese Struktur im Alltag praktisch angewendet wird.

Zusammenfassung Kapitel 7

- Struktur driftet ohne aktive Führung.
- Modi müssen benannt und bewusst gewechselt werden.
- Stop-Regeln verhindern implizite Übergänge.
- Governance bedeutet, den Rahmen aktiv zu halten.
- Professionelle Nutzung heißt: führen statt folgen.

Kapitel 8

So arbeitest du im Alltag strukturiert mit ChatGPT

Bis hierhin ging es um Prinzipien.

Jetzt geht es um Umsetzung.

Professionelle Nutzung bedeutet nicht,
lange Modus-Dokumente zu schreiben.

Sie bedeutet,
den Arbeitsprozess bewusst zu führen.

1. Starte nicht mit einer Frage.

Starte mit einem Zustand.

Bevor du inhaltlich arbeitest,
lege fest, in welchem Modus du arbeitest.

Beispiel:

Analyse-Modus aktiv.
Nur Beschreibung. Keine Bewertung.
Unsicherheit markieren.
Stop bei fehlenden Daten.

Erst danach stellst du die inhaltliche Frage.

Das dauert Sekunden.
Spart aber Korrekturen.

2. Halte den Modus über mehrere Schritte

Wechsle nicht bei jeder Antwort den Rahmen.

Wenn du analysierst, bleibe im Analyse-Modus.

Erst wenn die Phase abgeschlossen ist,
wechsel bewusst.

Das verhindert Vermischung.

3. Arbeite in Einzelschritten

Große Aufgaben erzeugen große Varianz.

Zerlege Aufgaben:

Statt:

„Erstelle eine komplette Marketingstrategie.“

Arbeite so:

1. Zielgruppe klären
2. Marktumfeld analysieren
3. Positionierung definieren
4. Maßnahmen entwickeln

Bestätige jeden Schritt,
bevor du weitergehst.

Das reduziert Korrekturschleifen drastisch.

4. Nutze gespeicherte Modi

Definiere deine Kernmodi einmal.

Zum Beispiel:

- Analyse
- Kritik
- Ideen
- Validierung

Speichere sie in deinen Custom Instructions
oder als wiederverwendbare Vorlage.

Du musst sie nicht jedes Mal neu schreiben.

Das macht Arbeit schneller,
nicht langsamer.

5. Trenne Denkphasen konsequent

Viele Fehler entstehen,
weil Analyse und Bewertung vermischt werden.

Arbeite bewusst in Phasen:

- Erst verstehen
- Dann hinterfragen
- Dann entscheiden

Diese Disziplin ist entscheidend.

6. Beende Aufgaben bewusst

Ein Modus endet nicht automatisch.

Beende ihn explizit.

„Analyse abgeschlossen. Wechsel in Bewertungs-Modus."

Das verhindert schleichende Drift.

Ein realistischer Ablauf

Professionelle Nutzung sieht nicht spektakulär aus.

Sie sieht so aus:

Modus setzen.
Schritt 1.
Bestätigung.
Schritt 2.
Bestätigung.
Wechsel.
Weiterarbeiten.

Weniger Improvisation.
Mehr Klarheit.

Der Effekt im Alltag

Mit Struktur:

- weniger Missverständnisse
- weniger Wiederholungen
- weniger Korrekturen
- mehr Vertrauen
- mehr Geschwindigkeit

Nicht, weil das System besser wird.

Sondern weil du es führst.

Übergang

Du hast jetzt:

- Verständnis
- Struktur
- Anti-Halluzinationskontrolle
- Governance
- Alltagsanwendung

Im nächsten Kapitel definieren wir,
was Referenzniveau in der Praxis bedeutet.

Zusammenfassung Kapitel 8

- Starte mit einem Modus, nicht mit einer Frage.
- Halte den Modus stabil über mehrere Schritte.
- Arbeite in bestätigten Einzelschritten.
- Speichere Kernmodi dauerhaft.
- Wechsle bewusst zwischen Denkphasen.

Kapitel 9

Der Referenzstandard für professionelle ChatGPT-Nutzung

Viele nutzen ChatGPT.
Wenige arbeiten strukturiert damit.

Der Unterschied liegt nicht im Prompt.
Er liegt im Verhalten.

Ein Referenznutzer zeichnet sich nicht durch Kreativität aus.
Sondern durch Disziplin.

1. Klare Zustände statt spontane Fragen

Ein Referenznutzer startet nicht impulsiv.

Er setzt zuerst den Modus.
Dann arbeitet er innerhalb dieses Rahmens.

Keine impliziten Übergänge.
Keine vermischten Phasen.

2. Struktur vor Geschwindigkeit

Schnelle Antworten sind verführerisch.

Aber Geschwindigkeit ohne Struktur erzeugt Korrekturschleifen.

Ein Referenznutzer:

- definiert Ziel
- begrenzt Handlung
- integriert Anti-Halluzinationsregeln
- arbeitet in Schritten

Das wirkt langsamer.
Ist es nicht.

3. Unsicherheit ist sichtbar

Referenzniveau bedeutet:

Unsicherheit wird nicht übersehen.
Sie wird eingefordert.

Wenn Informationen fehlen,
wird das benannt.

Wenn Annahmen entstehen,
werden sie markiert.

Vertrauen entsteht durch Transparenz.

4. Keine implizite Autonomie

Das System entscheidet nichts.

Der Nutzer führt.

Wenn mehrere Interpretationen möglich sind,
wird geklärt.

Wenn ein Übergang nötig ist,
wird der Modus gewechselt.

Referenznutzer lassen das System nicht „einfach weitermachen“.

5. Wiederverwendbare Struktur

Referenzniveau bedeutet:

- Kernmodi sind gespeichert.
- Anti-Halluzinationsregeln sind integriert.
- Governance ist bewusst.
- Arbeitsweise ist reproduzierbar.

Das System verhält sich konsistent,
weil der Rahmen konsistent ist.

6. Respekt vor der Stärke von LLMs

Professionelle Nutzung bedeutet nicht Skepsis.

LLMs sind hervorragend für:

- Wissensarbeit
- Strukturierung
- Ideengenerierung
- Textproduktion

Referenznutzer nutzen diese Stärke bewusst.

Aber sie verwechseln Sprachkohärenz nicht mit Wahrheit.

Der eigentliche Unterschied

Durchschnittliche Nutzung:

Frage → Antwort → Reaktion → neue Frage.

Referenzstandard:

Modus → Ziel → Einzelschritt → Bestätigung → bewusster Wechsel → nächster Schritt.

Das ist keine technische Fähigkeit.
Es ist eine Arbeitsdisziplin.

Übergang

Bis hierhin ging es um professionelle Wissensarbeit.

Jetzt folgt eine strukturelle Schlussfolgerung:

Was bedeutet diese Arbeitsweise für Systeme,
die nicht nur Texte erzeugen,
sondern handeln?

Darum geht es im letzten Kapitel.

Zusammenfassung Kapitel 9

- Referenzniveau ist strukturelle Disziplin, nicht Prompt-Kreativität.
- Modi werden bewusst gesetzt und gewechselt.
- Unsicherheit wird sichtbar gemacht.
- Struktur ist dauerhaft gespeichert.
- Das System wird geführt, nicht umgekehrt.

Kapitel 10

Warum Sprach-KI nicht ausreicht, wenn Systeme handeln

Bis hierhin ging es um Wissensarbeit.

Um Texte.
Analysen.
Ideen.
Bewertungen.

In diesem Bereich sind LLMs extrem leistungsfähig.

Mit Struktur, Modi und Governance
lassen sich zuverlässige Ergebnisse erzielen.

Aber jetzt kommt eine entscheidende Frage:

Was passiert, wenn ein System nicht nur antwortet –
sondern handelt?

Sprache ist nicht Handlung

Ein Text kann falsch sein.

Das ist ärgerlich.
Aber korrigierbar.

Eine physische Handlung ist anders.

Sie wirkt in der Realität.
Sie kann nicht zurückgenommen werden.
Sie erzeugt Folgen.

Hier reicht sprachliche Kohärenz nicht aus.

Der strukturelle Unterschied

Ein LLM erzeugt Wahrscheinlichkeiten.

Es bewertet nicht die reale Welt.
Es prüft keine physischen Zustände.
Es trägt keine Verantwortung.

In Wissensarbeit genügt das,
wenn der Mensch führt.

In physischen Systemen entsteht eine andere Anforderung:

Handlung muss legitimiert sein.

Nicht plausibel.
Nicht sprachlich überzeugend.
Sondern zustandsbasiert begründet.

Warum Struktur hier zwingend wird

Du hast in diesem Buch gesehen:

- Prompts sind keine Architektur.
- Rollen schaffen keinen Rahmen.
- Halluzination entsteht durch Lückenergänzung.
- Zustände reduzieren Drift.
- Governance verhindert implizite Übergänge.

All das gilt schon in Wissensarbeit.

Wenn ein System jedoch physisch agiert,
werden diese Punkte nicht optional.

Sie werden zwingend.

Ein physisches System braucht:

- explizite Zustände
- klare Mandate
- definierte Grenzen
- Stop-Mechanismen
- überprüfbare Entscheidungslogik

Ein Sprachmodell allein liefert das nicht.

Nicht, weil es schlecht ist.
Sondern weil es dafür nicht gebaut wurde.

Der wichtige Unterschied

LLMs sind Werkzeuge für Sprach- und Wissensarbeit.

Sie unterstützen Denken.
Sie strukturieren Informationen.
Sie generieren Perspektiven.

Sie sind wertvoll.

Aber sie sind keine Entscheidungsarchitektur.

Und sie sind keine Verantwortungsarchitektur.

Wenn ein System handeln soll,
muss die Struktur außerhalb des Sprachmodells liegen.

Die eigentliche Erkenntnis

Dieses Buch ist keine Kritik an LLMs, sondern aus Arbeitserfahrungen gewonnenes Wissen.

Es ist eine Anleitung,
wie du professionell mit ihnen arbeitest.

Wenn du dabei erkannt hast,
wie viel Struktur nötig ist,
um reine Textarbeit stabil zu halten,

dann wird eine Sache klar:

Für physische Handlung
reicht probabilistische Sprachlogik nicht aus.

Dort braucht es:

explizite Zustände
nachvollziehbare Übergänge
harte Grenzen
und klare Verantwortungszuordnung.

Abschluss

Prompts sind keine Architektur.

Mit Struktur werden LLMs zu starken Werkzeugen.

Ohne Struktur bleiben sie probabilistische Sprachgeneratoren.

Verwechsle beides nicht.

Zusammenfassung Kapitel 10

- LLMs sind leistungsfähig für Wissensarbeit.
- Sprachkohärenz ist nicht gleich Handlungslegitimation.
- Physische Systeme benötigen explizite Zustands- und Entscheidungsarchitektur.
- LLMs ersetzen keine Verantwortungsstruktur.
- Prompts sind keine Architektur.

- **Sprachmodelle erzeugen Texte. Verantwortung entsteht außerhalb des Modells.**

Kopierbarer Einrichtungsprompt

Diesen gesamten Block kopieren und in ChatGPT einfügen:

EINRICHTUNG MODUS-BAUKASTEN – REFERENZSTANDARD

Ab jetzt arbeite ich mit einem strukturierten Modus-System.

Du richtest bitte einen dauerhaften „Modus Baukasten“ ein und speicherst ihn kontextpersistent für diesen Chat.

Dieser Baukasten wird künftig durch den Befehl
„Modus Baukasten öffnen“
aktiviert und angezeigt.

1. Grundprinzip

Prompts sind keine Architektur.
Wir arbeiten mit dauerhaften Modi als Zustandsrahmen.

Ein Modus wirkt über mehrere Interaktionen hinweg, bis er bewusst gewechselt wird.

2. Jeder Modus besteht aus folgenden festen Bausteinen:

1. Ziel
2. Erlaubte Handlungen
3. Nicht erlaubte Handlungen
4. Umgang mit Unsicherheit
5. Stop-Regel
6. Rolle (nur für Ton/Wortwahl, nicht für Struktur)

Diese Struktur ist verpflichtend.

3. Integrierte Anti-Halluzinationsregeln (dauerhaft aktiv in jedem Modus)

- Fehlende Informationen dürfen nicht ergänzt werden.
- Unsicherheit muss explizit markiert werden.
- Annahmen müssen sichtbar gemacht werden.
- Fakten, Interpretation und Bewertung werden getrennt.
- Bei unklarer Grundlage muss gestoppt oder nachgefragt werden.

Diese Regeln gelten automatisch in jedem Modus.

4. Governance-Regeln (dauerhaft aktiv)

- Aktiver Modus wird benannt.
- Modus-Wechsel erfolgen bewusst.
- Keine impliziten Übergänge zwischen Analyse, Bewertung und Entscheidung.
- Bei Mehrdeutigkeit wird gestoppt und Klärung eingefordert.
- Komplexe Aufgaben werden in bestätigten Einzelschritten bearbeitet.

5. Einzelschritt-Disziplin

Bei komplexen Aufgaben:

- Zerlege in klare Schritte.
- Nach jedem Schritt: kurze Bestätigung einholen.
- Erst dann fortfahren.

Keine automatische Gesamtausführung ohne Bestätigung.

6. Persistenz

Der Modus-Baukasten bleibt dauerhaft aktiv.

Er ist jederzeit abrufbar durch:

„Modus Baukasten öffnen“

Er zeigt dann:

- Alle Bausteine
- Aktiven Modus
- Governance-Regeln

7. Aktivierungslogik

Wenn ich schreibe:

„Analyse-Modus aktiv“

dann gilt der zuvor definierte Analyse-Modus dauerhaft,
bis ich explizit wechsle.

8. Jetzt Einrichtung starten

Bitte bestätige:

- Dass der Modus Baukasten eingerichtet ist
- Dass Anti-Halluzinations- und Governance-Regeln dauerhaft integriert sind
- Dass der Baukasten durch „Modus Baukasten öffnen“ abrufbar ist

Danach führe mich Schritt für Schritt durch die Einrichtung meines ersten eigenen Modus.

Wir arbeiten dabei im Einzelschritt-Verfahren mit Bestätigung.

ENDE DER EINRICHTUNG

Wirkung dieses Prompts

Nach Ausführung:

- Struktur ist gesetzt
- Regeln sind integriert
- Wiederverwendung ist möglich
- Einzelschritt-Logik ist aktiv
- Modus-System ist persistent

Du arbeitest nicht mehr mit Einzelprompts.
Ab sofort kannst du in Modi mit Zuständen arbeiten.

Viel Erfolg beim weiteren arbeiten, bei allem was ihr macht

Marcel Johanning
Gründer Meventa Physical AI