

# Microsoft ordnet KI neu. Prompting bleibt relevant.

## Microsoft Copilot, Copilot Studio und Foundry

Orientierung zur praktischen Anwendung für Vertrieb, Marketing und Geschäftsführung.



Stand: 26. März 2026

© THINK&SELL elbe GmbH

# KI effektiv nutzen in Marketing und Vertrieb

[Die aktuellen Kurse](#)

Kurzfassung: Microsoft macht KI als System sichtbarer. Copilot steht für die Nutzeroberfläche, Copilot Studio für Low-Code-Agenten, Foundry für Modelle, Agentenbetrieb und Governance. Entscheidend ist deshalb nicht nur die Tool-Auswahl, sondern das passende Betriebsmodell.

### 1. Warum die Einordnung jetzt notwendig ist

Der Begriff Microsoft Copilot wirkt auf den ersten Blick wie der Name eines einzelnen Werkzeugs. In der Praxis steht er heute jedoch für mehrere Ebenen: für Nutzeroberflächen, für Low-Code-Agenten, für Plattformdienste und für spezialisierte Microsoft-Agents. Genau daraus entsteht in vielen Unternehmen Verwirrung. Wer nur von Copilot spricht, meint oft nicht dasselbe.

Microsoft selbst beschreibt die aktuelle Entwicklung als integriertes System mit vier verbundenen Säulen: Copilot Experience, Copilot Platform, Microsoft 365 Apps und AI Models. Für Unternehmen ist das ein wichtiges Signal. KI soll nicht mehr als Sammlung isolierter Funktionen verstanden werden, sondern als Teil einer durchgängigen Arbeits- und Betriebslogik.

#### 1.1 Was sich dadurch für Unternehmen ändert

Mit dieser Neuordnung verschiebt sich die Diskussion weg von der reinen Toolfrage. Wichtiger werden drei Punkte: Erstens, welche Aufgaben Anwender direkt in ihrer Arbeitsoberfläche erledigen sollen. Zweitens, welche Automatisierungen Fachbereiche selbst aufbauen dürfen. Drittens, welche KI-Lösungen unter Governance, Monitoring, Rollen- und Datenregeln produktiv betrieben werden müssen. Gerade im B2B-Umfeld entscheidet diese Unterscheidung darüber, ob KI als Hilfe im Alltag ankommt oder als neues Nebensystem stehen bleibt.

## 2. Was Copilot, Copilot Studio, Foundry und Azure Copilot Agents unterscheidet

Die Unterschiede liegen weniger in der Chat-Oberfläche als in Zweck, Zielgruppe und Betriebsmodell. Die folgende Übersicht hilft bei der Einordnung.

Ebene	Typische Rolle	Wann die Ebene sinnvoll ist
Copilot	Nutzernahe Produktivität	Wenn Mitarbeiter in bestehenden Anwendungen Unterstützung im Arbeitsfluss brauchen.
Copilot Studio	Low-Code für Fachbereiche	Wenn eigene Agents, Wissenszugriffe und Agent Flows ohne klassische Entwicklerplattform aufgebaut werden sollen.
Foundry	Plattform für IT und Entwicklung	Wenn Modelle, Runtime, Agentenbetrieb, Monitoring und Governance sauber gesteuert werden müssen.
Azure Copilot Agents	Microsoft-Spezialagenten für Azure	Wenn vorgefertigte Agents definierte Cloud-Operations-Aufgaben übernehmen sollen.

### 2.1 Copilot: produktive KI in der Nutzeroberfläche

Copilot steht aus Unternehmenssicht zuerst für den direkten Zugriff auf KI in bestehenden Arbeitskontexten. Anwender formulieren Fragen, fassen Inhalte zusammen, bereiten Entwürfe vor oder erhalten Unterstützung im jeweiligen Produkt. Der Fokus liegt auf Produktivität, nicht auf dem Betrieb eigener KI-Plattformen.

### 2.2 Copilot Studio: Low-Code für eigene Agents

Copilot Studio ist die grafische Low-Code-Umgebung, um eigene Agents und Agent Flows zu erstellen. Diese Ebene ist sinnvoll, wenn Fachbereiche oder Power User wiederkehrende Aufgaben strukturieren, Wissensquellen anbinden oder einfache Automatisierungen umsetzen wollen. Copilot Studio ist damit näher an der Fachanwendung als an einer klassischen Entwicklerplattform.

Wichtig ist die Abgrenzung zu Foundry: Copilot Studio ist nicht die spätere Entwicklungsstufe desselben Werkzeugs. Es ist eine andere Ebene mit anderem Schwerpunkt. Wo Geschwindigkeit, Konnektoren und visuelle Orchestrierung im Vordergrund stehen, ist Copilot Studio passend. Wo Modellwahl, Runtime, Deployment, Observability und Governance im Mittelpunkt stehen, beginnt Foundry.

### 2.3 Foundry: Plattform für Modelle, Agentenbetrieb und Governance

Foundry ist die Plattformschicht für Unternehmen, die KI nicht nur nutzen, sondern kontrolliert bauen und betreiben wollen. Im Foundry-Portal werden Projekte verwaltet, Modelle bereitgestellt, Agents gebaut und KI-Ressourcen überwacht. Dazu kommen APIs und SDKs für Entwickler sowie der Foundry Agent Service als managed Plattform zum Bauen, Deployen und Skalieren von Agenten.

Die Relevanz für Unternehmen liegt in der Betriebsfähigkeit: KI wird damit zum steuerbaren Artefakt mit Rollen, Richtlinien, Monitoring, Evaluierung und nachvollziehbaren Betriebsgrenzen. Genau das ist der Unterschied zwischen einer guten Demo und einer produktiven Lösung.

## 2.4 Azure Copilot Agents: fertige Spezialagenten statt Baukasten

Azure Copilot Agents gehören wiederum nicht zur Baukasten-Ebene, sondern zur Produkt-Ebene. Microsoft führt dafür spezialisierte Agents ein, die Aufgaben in Cloud Operations übernehmen, etwa für Migration, Deployment, Optimization, Observability, Resiliency oder Troubleshooting. Diese Agents zeigen, wohin die Entwicklung geht: weg von reiner Textausgabe, hin zu agentischer Ausführung innerhalb definierter Azure-Kontrollen.

Auch hier ist die Unterscheidung wichtig: Foundry Agent Service und Azure Copilot Agents sind keine alte und neue Bezeichnung für dasselbe Tool. Foundry Agent Service ist die Plattform zum Bauen und Betreiben eigener Agents. Azure Copilot Agents sind von Microsoft gelieferte Spezialagenten für konkrete Aufgabenbereiche.

## 2.5 Foundry Local: lokale Ausführung statt ausschließlicher Cloud-Nutzung

Ergänzend dazu erweitert Foundry Local die Architektur um lokale Inferenz auf dem Gerät. Damit adressiert Microsoft Szenarien, in denen Datenhoheit, Latenz oder Offline-Fähigkeit relevant sind. Für viele Unternehmen ist das kein Hauptthema im ersten Schritt. Es zeigt jedoch, dass die Foundry-Welt nicht nur für Cloud-Betrieb gedacht ist, sondern auch für Edge- und On-Device-Szenarien.

## 3. Was das für B2B-Unternehmen praktisch bedeutet

Für B2B-Unternehmen ist der Unterschied nicht akademisch. Er entscheidet über Verantwortlichkeiten, Investitionen und Einführungslogik. Wer nur eine produktive Unterstützung für Mitarbeiter sucht, denkt zuerst in Copilot. Wer Fachbereiche befähigen will, definierte Agents und Workflows selbst zu bauen, denkt in Copilot Studio. Wer KI unter stabilen Betriebs- und Governance-Anforderungen bereitstellen will, denkt in Foundry.

### 3.1 Wo Copilot und Copilot Studio gut passen

Copilot und Copilot Studio passen besonders dann, wenn der Mehrwert nah an den Anwendern entsteht: Zusammenfassungen, Entwürfe, Rechercheunterstützung, Wissenszugriff, Standardantworten oder einfache fachbereichsnahe Automatisierungen. Hier ist die Eintrittshürde niedriger, die Einführung schneller und der sichtbare Nutzen im Alltag oft früher erreichbar.

### 3.2 Wo Foundry den Unterschied macht

Foundry wird dann relevant, wenn ein Unternehmen mehr als nur die Oberfläche braucht: freie oder breitere Modellwahl, eigener Agentenbetrieb, spezifische Runtime-Anforderungen, tiefe Integration in Unternehmenslogik, kontrollierte Skalierung sowie observierbarer und governter Betrieb. Für strategische KI-Anwendungen reicht es nicht, einen Agent zu starten. Er muss mit Rollen, Daten, Regeln und Messbarkeit sauber in die Organisation eingebettet werden.

Gerade in Vertrieb und Marketing zeigt sich dabei ein wiederkehrendes Muster: Das technische Tool allein löst nicht den eigentlichen Engpass. Relevanter sind Kontextqualität, klare Formate, abgestimmte Freigaben und wiederholbare Prozesse. Ohne diese Grundlagen erzeugt auch ein gutes System nur wechselhafte Ergebnisse.

## 4. Welche Hilfestellung THiNK&SELL und KI-Training-B2B geben

Die Microsoft-Welt wird leistungsfähiger und zugleich komplexer. Genau deshalb ist ein strukturiertes Vorgehen entscheidend. THiNK&SELL und KI-Training-B2B setzen an unterschiedlichen, aber komplementären Punkten an.

## 4.1 THiNK&SELL: Strategie, Einführung und Prozessentwicklung

THiNK&SELL unterstützt Unternehmen dabei, KI nicht nur als neues Tool, sondern als veränderten Prozess zu betrachten. Dazu gehören Einführung, Rollen und Verantwortlichkeiten, Leitplanken für den Einsatz, Bewertung sinnvoller Use Cases sowie die Einbettung in Vertriebs- und Marketingabläufe. Kurz gesagt: weniger Tool-Wildwuchs, mehr Orientierung und Umsetzbarkeit.

Passende Anknüpfungspunkte sind insbesondere die strategische KI-Einführung und die schrittweise KI-Prozessentwicklung. Beide Leistungen helfen dabei, aus technischer Möglichkeit belastbare Arbeitsfähigkeit zu machen.

Weitere Informationen: <https://www.think-and-sell.eu/ki-einfuehrung/>

## 4.2 KI-Training-B2B: Methode, Assets und Transfer in den Alltag

KI-Training-B2B setzt auf den Teil, der im Alltag häufig über Erfolg oder Frust entscheidet: die Arbeitsfähigkeit der Teams. Statt nur Tools zu erklären, geht es um bessere Briefings, reproduzierbare Prompt-Strukturen, wiederverwendbare Assets und den Transfer in echte Vertriebs- und Marketingaufgaben. So entsteht aus punktueller Nutzung eine belastbare Routine.

Für Unternehmen ist das besonders relevant, weil sich Namen, Oberflächen und Funktionen von Plattformen weiter verändern werden. Wenn jedoch Methode, Qualitätskriterien und teamfähige Assets stehen, bleibt die Organisation handlungsfähig - auch dann, wenn sich der Werkzeugkasten weiterentwickelt.

Weitere Informationen: <https://ki-training-b2b.de/>

## 5. Fazit

Microsoft Copilot ist heute kein einzelnes Werkzeug mehr, sondern Teil eines größeren Systems. Copilot steht für den Nutzerzugang, Copilot Studio für Low-Code-Agenten, Foundry für Plattform, Betrieb und Governance und Azure Copilot Agents für spezialisierte Microsoft-Lösungen in konkreten Aufgabenfeldern. Wer diese Ebenen sauber trennt, kann Entscheidungen klarer treffen und KI sinnvoller einführen.

Für B2B-Unternehmen gilt deshalb: Nicht die Menge neuer Funktionen entscheidet über den Nutzen, sondern die passende Einordnung, der richtige Reifegrad und die Qualität der Umsetzung. Genau hier setzen THiNK&SELL und KI-Training-B2B als praktische Hilfestellungen an.

## Anhang: Quellen

Microsoft Learn: What is Microsoft Foundry?; Microsoft Learn: What is Microsoft Foundry Agent Service?; Microsoft Learn: Copilot Studio overview; Microsoft Learn: Choose an external model as the primary AI model; Microsoft Azure Blog: Announcing Azure Copilot agents and AI infrastructure innovations; Microsoft Blog: Announcing Copilot leadership update; Microsoft Azure Blog: Build and scale AI agents with Microsoft Foundry; Microsoft Learn: What is Foundry Local?; THiNK&SELL: KI-Einführung und KI-Prozessentwicklung auf [www.think-and-sell.eu](http://www.think-and-sell.eu).

*Hinweis: Diese Fassung wurde auf Basis einer Inhaltsvorgabe von THiNK&SELL mit Hilfe von ChatGPT erstellt*