



PUBLIC SAMPLE / PUBLISHED SEPTEMBER 2, 2026

Choosing an approach for grounded customer knowledge assistants.

A fictional Microsoft-focused MSP compares Copilot Studio, Microsoft Foundry Agent Service, and a custom Azure implementation.

CUSTOMER Fictional Microsoft-focused MSP	REPORT VERSION 1.0
RESEARCH CUTOFF September 2, 2026	EVIDENCE Public Microsoft sources

This sample demonstrates the Frontier AI Dev method. It is decision support, not security-certification, procurement, legal, or financial advice.

EXECUTIVE RECOMMENDATION

Use a two-tier standard. Treat fully custom Azure as an exception.

STANDARD

Copilot Studio

For rapid Microsoft 365, Teams, SharePoint, Dynamics, and Power Platform knowledge assistants.

ADVANCED

Foundry Agent Service

For Azure-native, application-integrated, private, extensible, or code-driven assistants.

EXCEPTION

Custom Azure

Only when a documented requirement remains unmet and funds justify the added lifecycle burden.

Microsoft Foundry Agent Service narrowly leads this fictional scenario because Azure-native extensibility and governance carry substantial weight. Copilot Studio becomes the preferred option when time to value and Microsoft 365 delivery dominate. There is no universal winner.

Scenario and assumptions

The fictional MSP wants a repeatable service for assistants that answer from customer-controlled sources, return citations, respect user access, reach Microsoft or custom channels, and support deployment, monitoring, evaluation, support, and cost allocation.

Custom Azure is an architecture category, not one Microsoft SKU. Its result depends on the selected framework, hosting, retrieval, identity, networking, and operational maturity.

Weighted decision matrix

Scores use a 1–5 scale. For skills and operations, a higher score means less MSP burden. These are Frontier AI Dev analysis under the fictional scenario, not Microsoft benchmarks.

CRITERION	WEIGHT	COPILOT STUDIO	FOUNDRY AGENT SERVICE	CUSTOM AZURE
Grounded-assistant fit	20%	4	5	5
Time to value	15%	5	4	2
Microsoft integration	15%	5	4	3
Extensibility	15%	3	5	5
Governance and security	15%	4	5	4
Skills burden	5%	5	3	1
Cost-model manageability	5%	3	3	2
Operational burden	10%	5	4	1
Weighted total	100%	85/100	88/100	67/100

Option profiles

OPTION 1

Microsoft Copilot Studio

Microsoft describes Copilot Studio as a graphical low-code environment for building agents and workflows, using connectors and publishing to Microsoft and external channels. [S1] Supported grounding patterns include SharePoint, Dataverse, uploaded files, websites, and connector-accessible enterprise data. [S2]

Best fit: Microsoft 365 and Teams assistants, SharePoint Q&A, Dynamics and Power Platform scenarios, and connector-led workflows where rapid configuration matters.

Watch: documented source limits, retrieval-control constraints, lifecycle requirements, and Microsoft's warning that disabling ungrounded answers is not an absolute guarantee against general model knowledge influencing an answer. [S2] [S10]

OPTION 2

Microsoft Foundry Agent Service

Microsoft Foundry Agent Service is the current official name for the managed platform supporting prompt agents, hosted agents, and application-owned patterns through the Responses API. [S4] Microsoft lists Azure AI Studio and Azure AI Foundry as previous brands. [S5]

Best fit: Azure-native custom applications, agents requiring code and tools, private environments, and MSPs that want a managed runtime without surrendering application-level extensibility.

Watch: model, tool, region, and feature availability; several-resource standard setups; additional private-network complexity; and continuing product evolution. [S6] [S7]

OPTION 3

Custom Azure implementation

A custom approach combines selected application hosting, models, retrieval, identity, networking, monitoring, and MSP-owned orchestration. Microsoft's RAG guidance separates ingestion, chunking, enrichment, indexing, retrieval, and evaluation into distinct design responsibilities. [S17]

Best fit: differentiated products, unusual retrieval or orchestration, strict UX and API control, or requirements that managed products cannot satisfy.

Watch: the MSP owns the largest security, reliability, evaluation, deployment, dependency, isolation, recovery, and support surface. Microsoft's basic reference architecture is explicitly not a production architecture. [S19] [S20]

Decision sensitivity

SCENARIO	LIKELY LEADER
Teams or SharePoint employee assistant delivered in weeks	Copilot Studio
External portal with custom UX and APIs	Foundry Agent Service
Advanced bespoke retrieval with deterministic control	Custom Azure or a Foundry hosted agent
Small customer with limited Azure skills	Copilot Studio
Regulated Azure workload requiring private networking and customer-owned state	Foundry standard setup, subject to validation

Cost-model conclusion

A responsible dollar comparison requires region, currency, contract, model, deployment type, traffic, response length, retrieval, concurrency, availability, retention, networking, and existing Microsoft entitlements. This sample therefore compares cost structures rather than inventing totals.

- **Copilot Studio:** Copilot Credits with feature-dependent consumption and licensing conditions.[S23]
- **Foundry Agent Service:** model inference, tools, hosted compute where used, state resources, retrieval, telemetry, and networking.
- **Custom Azure:** separate model, retrieval, hosting, storage, network, monitoring, resilience, engineering, security, and support costs.

Questions to validate before selection

1. What is the primary channel and user identity type?
2. Which sources are authoritative?
3. Must access respect each user's source permissions?
4. Are citations and abstention mandatory?
5. What volume, freshness, language, and metadata are required?
6. Must state remain in customer-owned resources?
7. Are private endpoints and restricted egress mandatory?
8. Which regions, models, and preview features are allowed?
9. How will prompt injection be tested?
10. What telemetry can contain customer content?
11. What latency and availability are promised?
12. Who owns outages and degraded answers?
13. Which licenses does the customer already hold?
14. How will cost be tested under replayed traffic?

Recommended next step

Build one representative assistant in Copilot Studio and one as a Foundry prompt agent using the same source set and evaluation questions. Compare grounding, citations, security trimming, abstention, prompt-injection resistance, freshness, latency, deployment, rollback, and workload cost. Test a Foundry hosted agent before accepting the lifecycle burden of a fully custom runtime.

Primary source register

All sources were retrieved September 2, 2026.

1. [Overview — Microsoft Copilot Studio](#). Microsoft. Updated August 22, 2026.
2. [Knowledge sources summary — Microsoft Copilot Studio](#). Microsoft. Updated August 27, 2026.
3. [What is Microsoft Foundry Agent Service?](#) Microsoft. Updated August 27, 2026.
4. [What is Microsoft Foundry?](#) Microsoft. Updated August 27, 2026.
5. [Foundry Agent Service quotas, limits, and regions](#). Microsoft. Updated August 28, 2026.
6. [Set up standard agent resources](#). Microsoft. July 9, 2026.
7. [Quotas and limits — Microsoft Copilot Studio](#). Microsoft. Updated August 4, 2026.
8. [Design and develop a RAG solution on Azure](#). Microsoft. Updated July 2, 2026.
9. [Baseline Microsoft Foundry Chat reference architecture](#). Microsoft. Updated August 18, 2026.
10. [Basic Microsoft Foundry Chat reference architecture](#). Microsoft. Updated August 18, 2026.
11. [Standard harness licensing — Microsoft Copilot Studio](#). Microsoft. August 3, 2026.