SPSync

Microsoft AI stack for SharePoint lists (PDF)

A dated map of Copilot connectors, Work IQ, Copilot Studio knowledge, Agent Builder and MCP: what each reads from a SharePoint list and its limits.

Your manager wants "an agent that knows the Requests list", and the first search turns up seven Microsoft products that all mention SharePoint. Copilot connectors, Work IQ, Copilot Studio knowledge, Agent Builder, declarative agents, MCP servers and agents in SharePoint each have a Learn page, and none of them says how many rows of your list it will actually read. This page puts them side by side. For each one it gives what it reads from a list, the limit Microsoft publishes, the licence it needs and where a SQL copy of the list fits.

Every status and limit below was read on Microsoft Learn on 23 October 2026. These products are renamed and moved between preview and general availability every few months, so check the linked page before you commit to one. We review this map each quarter.

The map at a glance

Surface | What it reads from a list | Published limit | Licence | Status on 23 Oct 2026 | Where a SQL replica fits

Copilot Studio list knowledge | List rows, ingested into Dataverse and indexed | First 2,048 rows queried; 15 lists, 35,000 rows a list | Copilot Studio | Documented, no preview label | Replaces it once counts matter

Copilot Studio real-time knowledge (Azure SQL, Azure AI Search) | Table and column names only; queries run live | None stated | Copilot Studio | Preview | The replica is the source

Agent Builder list knowledge | One list per agent | 20,000 items and 50 MB of raw text | Varies by licence | Documented | Through a Copilot connector over the replica

Copilot connectors (synced) | Data outside Microsoft 365; no SharePoint Online list connector | 40-second query timeout on the SQL connectors | Microsoft 365 for Search; Copilot add-on for Copilot grounding | Azure SQL and SQL Server documented; PostgreSQL preview | The SQL connector indexes replica rows

Work IQ MCP | Microsoft 365 data through generic tools over /sites/ paths | 25 items a call by default, 100 at most, no paging | Usage-based billing | Work IQ MCP out of preview; Copilot Studio integration preview | None; keep it for mail, meetings and files

MCP tools in Copilot Studio | Whatever the server exposes | Streamable transport only | Copilot Studio | Documented | Point a database MCP server at the replica

Agents in SharePoint | Sites, pages and document libraries | Lists not named | Copilot licence or pay-as-you-go | Documented | None

Can a Copilot connector index a SharePoint list?

Not directly. Microsoft Learn's Copilot connectors overview describes connectors as the way to bring in data from beyond Microsoft 365, and the Microsoft-built connectors gallery has a SharePoint Server connector for on-premises sites and no connector for SharePoint Online lists. A

list in SharePoint Online is already Microsoft 365 content, so a search for "graph connector sharepoint list" mostly finds people who want something the gallery does not offer.

The gallery does have Azure SQL and Microsoft SQL Server connectors, and a PostgreSQL connector in preview. That is the route for a replica. The Azure SQL and SQL Server connector page says the connector indexes the rows a SQL query of your choosing returns, with every selected column becoming a property you can mark searchable. The query has to finish within a 40-second timeout. Row-level access comes from access control columns you add to the query, holding user or group ids, and the Azure SQL database has to sit in the same Microsoft Entra tenant as Microsoft 365.

The deployment page asks for a watermark column for the full crawl, an optional incremental crawl on a modified-date column, and an optional soft-delete column with the value that marks a row as deleted. By default the incremental crawl runs every 15 minutes and the full crawl every day. An SPSync table has the columns this wants. sp_modified is the watermark, sp_item_id is the key, and a view can turn sp_deleted_at into the deleted flag the connector asks for. Microsoft's licensing table says synced connectors reach Microsoft Search on any Microsoft 365 plan, need the Microsoft 365 Copilot add-on to ground Copilot, and reach agents only with a Copilot Studio licence.

A connector feeds search, so "show me the open requests for Leeds" works well, while "how many requests are open" still depends on how many rows the retrieval step hands the model. For counts, use one of the query routes further down.

Copilot Studio knowledge sources for SharePoint lists

Copilot Studio has two kinds of knowledge that matter here. The first is the list itself. Learn's Add SharePoint as a knowledge source says selected lists are ingested into Dataverse and indexed, that users sign in with their own SharePoint credentials, and that it is best to keep to 10 lists per agent. The quotas and limits page, updated the day before we checked it, holds the numbers that matter: list queries only return data from the first 2,048 rows, an agent takes up to 15 lists and 35,000 rows a list within 120,000 rows in total, a list view cannot be the source, and a default view with more than 12 lookup columns is not supported. A list that passes setup can still be answered from its first 2,048 rows, with no warning in the reply.

The second kind is what Learn calls Power Platform real-time knowledge connectors. The supported list on 23 October 2026 includes Azure SQL and Azure AI Search, and the page is marked as preview and prerelease documentation. Microsoft indexes only table and column names, and each question runs against the source at runtime with the signed-in user's token. With a replica in Azure SQL the database does the counting over every row, and with embeddings written to Azure AI Search the agent can search by meaning. SPSync writes both, and generates embeddings on sync for Azure AI Search. The Copilot Studio grounding use case walks through the setup, and the three routes post compares it with a flow tool and a Dataverse copy.

Learn's licensing page says a standalone Copilot Studio subscription connects to any data through premium connectors, and that usage is measured in Copilot Credits.

Does Work IQ read SharePoint list items?

Partly, and in small pieces. Work IQ is Microsoft's name for the intelligence layer behind Copilot, and Work IQ MCP is its MCP server. The Agent 365 MCP server options page describes it as one endpoint with 10 generic tools that discover resource paths and schemas at runtime, and says Work IQ MCP is out of preview as part of the Work IQ API, while the Work IQ integration in Copilot Studio is still in preview. The Work IQ API overview lists what it reasons over: mail, meetings, documents in OneDrive and SharePoint, Teams messages, people, Planner plans and enterprise search results. Lists are not on that list.

The fetch tool can still reach a list, because the tool reference allows /sites/ paths by default. The same page sets the limit that decides it for a large list. Collection results get a default $top of 25 items, capped at 100, and the $skip and $skiptoken parameters that page through a longer list are blocked. An agent using Work IQ can read a list item it already knows the id of. It cannot walk a 9,000-row list.

Setup is a tenant decision. Learn's Enable your tenant for Work IQ page asks for a usage-based billing plan in Copilot Studio with an Azure subscription, and a Global Administrator for the one-time setup. The Copilot Studio page adds that Work IQ there bills in Copilot Credits and is read-only unless an administrator turns on write operations.

You will also find an older SharePoint MCP server in the Copilot Studio tool catalogue. Its reference page lists listListItems, getListItem and tools that create, update and delete items, with file operations capped at 5 MB. Learn marks it as preview and says these application-specific servers are kept for backward compatibility and should not be used in production or new projects.

MCP in Copilot Studio over a list or a database

Copilot Studio's MCP support takes the tools and resources a server publishes and needs generative orchestration turned on. The page on connecting an existing server says only the Streamable transport is supported, SSE support ended after August 2025, and authentication is an API key or OAuth 2.0.

That makes MCP the general way to give a Copilot Studio agent a database. SPSync ships no agent. You point a database MCP server at the replica and give it a login that can only read. SPSync's own spsync mcp serve is a read-only MCP server over the replica, but its HTTP transport has no authentication, so it needs something that authenticates in front of it before a hosted agent calls it. For Azure SQL or SQL Server, Microsoft's own SQL MCP Server, part of Data API builder from version 1.7, exposes only the tables and views you list in its configuration, with permissions per role, and by design does not turn questions into free-form SQL. For PostgreSQL, any Postgres MCP server that runs under a read-only role does the same job. The tool sees typed tables where a

lookup is a key column and a person joins to a users table, which is the schema a list cannot give it.

Declarative agent and Agent Builder SharePoint list knowledge

Agent Builder takes a list, one at a time. Learn's Agent Builder knowledge page allows one SharePoint list per agent with a maximum of 20,000 items and 50 MB of raw text. Past either limit the agent truncates the list and says so in its reply. Attachments are not indexed, a site added as knowledge does not bring its lists with it, and Learn notes that knowledge sources differ by licence. The same page lists Copilot connectors as an Agent Builder source with no documented numerical limit, which is how a list larger than 20,000 items, indexed from its replica by the SQL connector above, can reach an Agent Builder agent.

In a hand-written manifest the picture is less clear than the property names suggest. The declarative agent manifest reference (schema 1.9, updated 22 October 2026) has a list_id field under items_by_sharepoint_ids, and describes it as the identifier for a document library, used with unique_id to scope a folder or file. Read it as a way to scope files. The manifest's GraphConnectors capability, which takes a connection_id, is the documented way to point a declarative agent at connector content, including a SQL connector over the replica.

What agents in SharePoint read

Learn's admin page, Get started with agents in SharePoint, says these agents find information on SharePoint sites, pages and document libraries, and answer from the data each user can already open. Lists are not named. A user needs a Copilot licence to create one; to use one, a licence or a tenant with pay-as-you-go billing is enough. For questions about a site's documents they are the quickest thing to set up. For questions about the rows of a list, use one of the routes above.

Which one should you start with?

Start from the questions people will ask. If they name one item, a lookup such as "what is the status of request 4821", list knowledge in Copilot Studio or Agent Builder answers it with SharePoint permissions included and nothing else to run. If they ask how many, which building has the most, or anything that joins two lists, the list routes run out at 2,048 or 20,000 rows, and Work IQ at 100 items a call. Those questions need a database that sees every row, reached through Azure SQL real-time knowledge or a database MCP server. If you want the rows to turn up in ordinary Copilot answers and Microsoft Search, index the replica with the Azure SQL or SQL Server Copilot connector.

Our AI on SharePoint data architecture guide sets out the layers behind this: SharePoint as the system of record, one reader, one replica and as many agents as you need.

What the replica does not change

SPSync copies list items and their columns, and library metadata. It does not copy file contents, so

document text stays with the SharePoint routes. It reads one way: SharePoint remains the system of record and nothing an agent sends reaches the list. Two-way sync is designed and not shipped. SharePoint item permissions are not carried into the replica, so access is set in the database with logins, roles, row-level security, or the access control columns of a Copilot connector query. Embeddings on sync are written for Azure AI Search and pgvector only. The replica is current to the sync schedule, every five minutes by default.

Full, up to date version: https://spsync.io/resources/microsoft-ai-stack-for-sharepoint-lists/